A Mission Planner to Improve the Cost-effectiveness of Autonomous Marine Vehicle Deployments

by

Fletcher Thompson (B.Eng., Naval Architecture)

National Centre for Maritime Engineering and Hydrodynamics

Australian Maritime College

Primary Supervisor: Dr. Damien Guihen

Co-supervisor: Dr. Gregor MacFarlane

Research Adviser: Dr. Roberto Galeazzi

Submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy (Marine Engineering)

University of Tasmania, February, 2019

Statement of Co-Authorship

The following people and institutions contributed to the publication of work undertaken as part of this thesis:

Candidate: Fletcher Thompson, National Centre for Maritime Engineering and Hydrodynamics
Author 1: Damien Guihen (Supervisor), University of Tasmania
Author 2: Roberto Galeazzi, Technical University of Denmark

Author details and their roles:

Paper 1, Review of Mission Planning for Autonomous Marine Vehicle Fleets:
Located in chapter 2
Candidate was the primary author and with author 1 contributed to the conception and design of the research.
Candidate contributed almost all the written work (>95%).
Author 1 contributed to the analysis and interpretation of the work by critically revising the paper, and contributed two graphical representations of the research (figure 1 and table 2).

Paper 2, Robust Mission Planning for Long Term Operation of a Fleet of Autonomous Marine Vehicles (Preprint):
Located in chapter 4
Candidate was the primary author and with author 2 contributed to conception and design of the research.
Candidate contributed most of the written work (>90%).
Author 2 contributed section 2.3: "Energy Consumption Prediction" as well as contributed with technical feedback and critical review of the candidate's contribution.

Paper 3, Field Trials of an Energy Aware Mission Planner Implemented on an Autonomous Surface Vehicle (Preprint):
Located in chapter 6
Candidate was the primary author and with author 1 and 2 contributed to conception and design of the research.
Candidate contributed almost all the written work (>95%).
Author 1 contributed to the analysis and interpretation of the work by critically revising the paper.
Author 2 contributed with technical feedback and critical review of the candidate's contribution.

We the undersigned agree with the above stated "proportion of work undertaken" for each of the above published (or submitted) peer-reviewed manuscripts contributing to this thesis:

Signed:	_____	_____
	Damien Guihen *Supervisor* *National Centre for Maritime Engineering and Hydrodynamics* *University of Tasmania*	*Shuhong Chai* *Head of School* *National Centre for Maritime Engineering and Hydrodynamics* *University of Tasmania*
Date:	21/02/2019	21/02/2019

Declaration of Originality

This thesis contains no material which has been accepted for a degree or diploma by the University or any other institution, except by way of background information and duly acknowledged in the thesis, and to the best of my knowledge and belief no material previously published or written by another person except where due acknowledgement is made in the text of the thesis, nor does the thesis contain any material that infringes copyright.

Authority of Access

This thesis may be made available for loan and limited copying and communication in accordance with the Copyright Act 1968.

Statement regarding published work contained in thesis

The publishers of the papers comprising chapter 2 hold the copyright for that content and access to the material should be sought from the respective journals. The remaining non-published content of the thesis may be made available for loan and limited copying and communication in accordance with the Copyright Act 1968.

Signed: _____

Fletcher Thompson
Candidate
National Centre for
Maritime Engineering
and Hydrodynamics

Date: 22/02/2019

Acknowledgements

I'm not sure how I've managed to sit on a stone for three years, but it occurs to me now, while I stretch my legs, that several dear people have helped me through the process in their own ways.

Thanks be to my parents, Sharen and Cameron Thompson, who are the biggest cheerleaders and morale-boosters I know. Special thanks to Cameron for rowing after rogue robots. Special thanks to Sharen for putting up with him.

To Damien Guihen, my primary supervisor, for keeping things in perspective without losing focus. To my co-supervisor, Gregor MacFarlane, for providing sound navigational advice. And to Shinsuke Matsubara, who settled me on the stone and was responsible for the stone metaphor in the first place.

To Roberto Galeazzi, who welcomed a stranger from the other side of the planet and has sharpened my knowledge of marine systems control every time we collaborate. To the rest of the fine folks at DTU's Department of Automation and Control, who facilitated an excellent research stay.

To Mike, Jock and Dags of the venerable AMC technical staff team, who have helped me with countless jobs over the years. And to Kirk SPCY and Ross from the towing tank staff, who helped me build the best possible autonomous self-propelled watertight box.

To the Wilsons, types K and S, for keeping the rain off my head.

Finally to Heidi Stenkvist, who is near even when far, a master of poster design and the opposite of an empty pocket. Now you're just a stone's throw away.

Abstract

Operators and end users of Autonomous Marine Vehicles (AMVs) value mission reliability (i.e. the probability that the AMV satisfies mission goals). Operators traditionally use conservative safety margins when planning missions for AMVs to reduce the risk of uncertainty affecting mission success. Although pragmatic, the policy is largely based on user experience and runs the risk of underusing or overextending the vehicle in missions where experience is lacking. Quantifying components of uncertainty improves mission plans by adding safety margins more closely attuned to both task completion and vehicle survivability. Mission uncertainty is influenced by how well the mission plan predicts the vehicle's performance in the marine environment. Current planners formulate missions as a schedule of timed tasks and only briefly consider the performance prediction in terms of time. A potential solution is to plan with energy consumption and energy capacity as the planning constraints. Unlike the current standard marine vehicle temporal planners, energy planning anticipates loadings on the AMV as it fulfils assigned tasks. In multi-vehicle missions, plans are made by allocating and scheduling tasks to vehicles based on their energy capacity. The task schedules can be optimised to minimise energy consumption and maximise the number of tasks completed. Based on the plan's energy prediction, the measured energy consumption of deployed vehicles can be evaluated to determine if correction is necessary. Energy planning maintains survivability by adhering to vehicle battery constraints and increases the operator's confidence in the plan as it has been thoroughly quantified, meaning that the safety margins governing the vehicle's overall exposure to risk are appropriately tuned. This thesis describes the development of a two-stage AMV Energy Planning Framework (EPF). In the first stage, the EPF automates the process of formulating the planning problem from mission data specified by the operator and then solves it to provide energy efficient plans for each vehicle in the fleet. The EPF formulates the mission data into the Team Orienteering Problem (TOP) and uses Monte Carlo simulation on the vehicle dynamic model to obtain the expected energy consumption for each possible edge in the TOP graph. The solver, a discrete implementation of particle swarm optimisation, was evaluated

using TOP testing data and was found to generate near optimum plans with resource scarcity (i.e. AMV fleets containing up to 4 members). The full first stage of the EPF was tested on a dataset from a wind turbine array inspection mission and was shown to generate concurrent mission plans for multiple underwater vehicles. The second stage occurs during deployment, where a mission supervisor agent onboard the vehicle monitors the progress of the plan relative to the vehicle's measured energy consumption. The supervisor decides if the current plan is likely to be completed and, if not, decides on a recourse action according to policy parameters specified by the operator. A prototype AMV was developed to perform missions provided by the first stage of the EPF and the results were used to evaluate the second stage. During deployments of the AMV in a lake environment, it was demonstrated that manipulation of the supervisor policy parameters resulted in encouragement of conservative or risky behaviours. A conservative policy caused the vehicle to return home sooner, completing tasks along the way. In contrast, a risky reliability parameter caused the vehicle to persist with the original mission plan for longer. By providing an accurate model of the vehicle and environment, operators using the EPF will be able to plan efficient missions tailored to the capabilities of their vehicles. Accurate models are not always available, so a data-driven Long Short-Term Memory (LSTM) neural network model was proposed to forecast vehicle energy consumptions based on a learned vehicle model. Conservative or risk-taking behaviour policies can be specified to influence task completion by the AMV without stranding it in an irrecoverable position. The result is a nuanced method for planning efficient and safe missions for AMVs. Operators using an EPF approach will be able to do more with their AMVs per mission, increasing the cost-effectiveness of expeditions.

Contents

List of Figures

List of Tables

List of Acronyms

3DR 3D Robotics

AI Artificial Intelligence

AMV Autonomous Marine Vehicle

APF Artificial Potential Field

ASV Autonomous Surface Vehicle

AUV Autonomous Underwater Vehicle

BIIMAPS Blackboard Integrated Implicit Multi-Agent Planning Strategy

CARACaS Control Architecture for Robotic Agent Command and Sensing

CaSHMI Control Station Human-Machine Interface

COLREGS International Regulations for Prevention of Collisions at Sea

C-SLAM Cooperative Simultaneous Localisation and Mapping

CV Complex Velocity

DC Direct Current

DCOP Distributed Constraint Optimisation Problem

Dec-POMDP Decentralised POMDP

DStPSO Discrete Strengthened Particle Swarm Optimisation

EKF Extended Kalman Filter

EPF Energy Planning Framework

EUROPA Extensible Universal Remote Operations Planning Architecture

EUROPA2 See EUROPA

EUROPtus See EUROPA

FSM Finite State Machine

GPS Global Positioning System

GUI Graphical User Interface

HiDDeN High level Distributed Decision

HMI Human-Machine Interface

HTN Hierarchical Task Network

HSTS Heurisitc Scheduling Testbed System

IA Instant Assignment

IMU Inertial Measurement Unit

IxTeT Indexed Time Tables

KaRL Knowledge and Reasoning Language

LA-DCOP Low-communication Approximate DCOP

LHS Left Hand Side

LOS Line of Sight

LSTM Long Short-Term Memory

LSTS Laboratory of Underwater Systems and Technology

MacDec-POMDP Macro-Action Decentralised POMDP

MADARA Multi-Agent Distributed Adaptive Resource Allocation

MAPGEN Mixed-Initiative Activity Plan Generation

MASSMO Marine Autonomous Systems in Support of Marine Observations

MBARI Monterey Bay Aquarium Research Institute

MDP Markov Decision Process

MOOS-IvP Mission Oriented Operating Suite - Interval Programming

MR Multiple Robot

MRS Multi-Robot Systems

MT Multiple Task

NASA National Aeronautics and Space Administration

NDDL New Domain Definition Language

NECSAVE Network Enabled Cooperation System of Autonomous Vehicles

NTNU Norwegian University of Science and Technology

OP Orienteering Problem

OP-SW Orienteering Problem with Stochastic Weights

OR Operations Research

PCV Partial Complex Velocity

PDDL Planning Domain Definition Language

PID Proportional Integral Derivative

POMDP Partially Observable Markov Decision Processes

PSO Particle Swarm Optimisation

PSR Point of Safe Return

R4SA Robust Real-time Reconfigurable Robotics Software Architecture

RC Radio Controller

RCPSP Resource Constrained Planning and Scheduling Problem

REMORA Reconfigurable Modular Robotic System for Aquatic Environment

RHS Right Hand Side

RKDP Runge-Kutta Dormand-Prince

RMSE Root-Mean-Squared Error

RNN Recurrent Neural Networks

ROS Robot Operating System

ROV Remotely Operated Vehicle

RPE Relative Percentage Error

RPM Revolutions Per Minute

RVNS Reduced Variable Neighbourhood Search

SAA Sample Average Approximation

SLAM Simultaneous Localisation and Mapping

SR Single Robot

ST Single Task

STRIPS Stanford Research Institution Problem Solver

Swarm-GAP Swarm - General Assignment Problem

SWARMs Smart and networking underWAter Robots in cooperation Meshes

TA Time Extended Assignment

TOP Team Orienteering Problem

T-REX Teleo-Reactive Executive

TSP Travelling Salesman Problem

UAV Unmanned Aerial Vehicle

USV Unmanned Surface Vehicle

UTM Universal Transverse Mercator

VNS Variable Neighbourhood Search

Chapter 1

Introduction

1.1 Motivation

Marine robots are the latest in the long line of tools developed to assist humans in safely accessing and performing work in a historically dangerous and costly workplace: the marine environment. They are specialised vehicles designed to achieve tasks that in the past have been too costly, dangerous, or impractical for humans. Examples of tasks include the inspection of fisheries at high latitudes with underwater vehicles (Karimanzira et al., 2014), deployment of unmanned surface vehicles to collect offshore floating rubbish (Chrissley et al., 2017), autonomous ships for disaster response and casualty detection (Martins et al., 2013), deployment and inspection of subsea cables (Ferguson et al., 1999), detection and disposal of sea mines (Nguyen et al., 2008), and under-ice bathymetry mapping (Kaminski et al., 2010).

Initially, marine robots were controlled remotely by human operators, referred to as Remotely Operated Vehicles (ROVs). Now, platforms capable of autonomously following user specified paths, automatically detecting and avoiding obstacles, handling unexpected hardware faults, and creating their own plans to achieve user specified goals are in active use. Robust autonomous vehicle platforms that are capable of hours of independent operation are available off-the-shelf, and can be outfitted by operators to handle a variety of payloads suitable for a wide area of applications.

With the growing demand for AMVs comes increasingly complex and novel mission profiles. Such missions require persistent or near-persistent autonomy (operating independently for very long amounts of time), causal inference of missions with interdependent components, multi-vehicle cooperation, and adaptation to dynamic conditions or objectives. All of these require-

ments depend upon reasoned models of the environment and its constraints, the cause-effect relationship between tasks and the fulfilment of objectives (or goals), and how a vehicle's capabilities (or its available actions) contribute to the completion of tasks. These models are reasoned, analysed, and developed in the sub-field of artificial intelligence research called automated planning and scheduling (also referred to as Artificial Intelligence (AI) planning).

The purpose of AI planning is to enable automata (e.g. marine robots and software programs) to generate a plan of scheduled actions which, when executed, will obtain a desired state configuration. The "state" is an abstract term used to describe the domain of interest. For example, the state for a game of chess could be represented by the location of pieces on a chessboard and who's turn it is. The effect of actions changes the state (moving pieces around the board in legal manoeuvres). The state can also be expanded to include a notion of continuous time space (important in chess games where there are turn-length constraints, or when the history of the state contains important information regarding the current and future states). AI planning uses a mathematical model of these action-state effects, referred to in the literature as a "planning domain" or "action language", as the problem space for a solver component to find a feasible plan within. Just as chess can be represented with a planning domain, there exists a planning domain for the marine environment. However, the planning domain of the marine environment is non-trivial to model: it is partially observable (the entire state is not available to the planner), time-varying, and stochastic. Still, planning domain models have been developed to allow AMV planning agents to produce feasible plans.

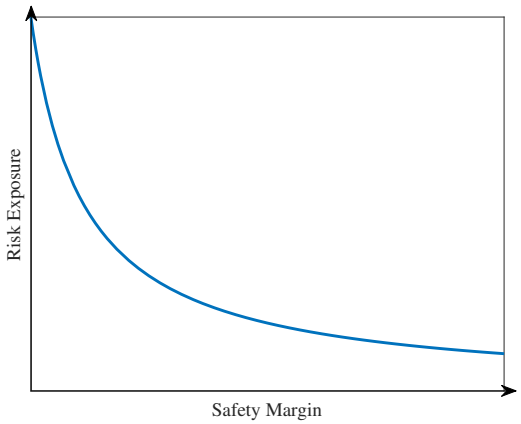
AI planning enables AMVs to schedule survival-critical activities (e.g. recharging, data-offload, maintenance) which extends their autonomy persistence. The planning domain is used as a basis for the modelling of cause-effect relationships for a mission, and enables the allocation and scheduling of actions for specific vehicles in fleet operations. Furthermore, the formation of a marine platform ontology (Bermudez et al., 2006, Li et al., 2017) presents the opportunity for a planning domain to *learn* new cause-effect relationships based on previous experience. At its core, planning is about making predictions on the outcomes of actions in order to achieve desired goals. Some of the fundamental problems in planning for robots can be linked with the same problems that are found in human planning.

When humans make plans, we create our own model of the world and use it as a testing ground to create and schedule actions that will change the state of the model to suit our goals. We make predictions on how the world model changes with given actions at given times, and then connect these actions through formal logic to obtain a plan. We are faced with two problems.

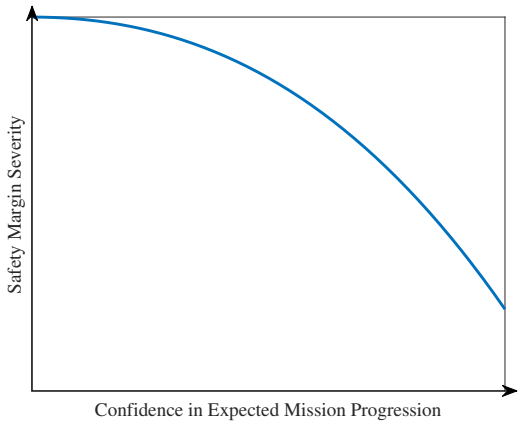
The first is that the outcomes of actions on the world model are uncertain, especially so in time-varying domains, and when there are multiple agents executing plans that are uninformed of one another's intentions. The second, which is affected by the first, is the so-called "planning fallacy" (Kahneman and Tversky, 1977). The planning fallacy has revealed that we tend to have an optimism bias when estimating the amount of time it takes to complete an action. The planning fallacy may seem like it only pertains to humans, but an AI planning agent may inherit its designer's flaws.

In the marine robot planning domain, engineers tend to introduce conservative safety margins on their vehicles in order to avoid directly confronting these planning problems. Glossing over these problems in this manner is expedient and still allows the vehicle to be deployed, but the end-user incurs significant opportunity costs for the sake of safety margins. Consider the figures in 1.1. Figure 1.1a illustrates a critical relation between the level of risk the operator expects the vehicle to be exposed to, given the controlled specification of safety margins. Operators mitigate risks during the execution phase via fail-safe (safety margin) strategies implemented on the vehicle. These fail-safe strategies can be conditional checks in the mission plan such as "if leak detected, then surface", or margin-based approaches such as planning with a portion of the vehicle's endurance removed as an emergency reserve.

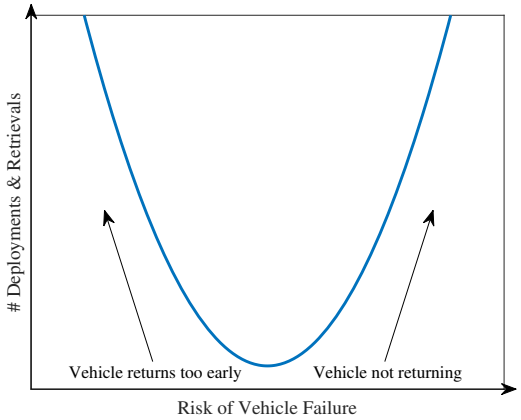
The operator must ensure that there are enough fail-safe checks and that the margins are tight enough so as to minimise catastrophic consequences while still allowing the vehicle to make tangible progress. The level of constriction in the design of these safety margins is a reflection of the operator's confidence in the expected vehicle response and behaviour in the environment, leading to the relationship in figure 1.1b. As in most risk-assessment situations, AMV operators need to quantify the level of uncertainty that the vehicle faces for a given mission profile so that appropriate safety margins for the mission can be designed. If the operator does not have enough information to quantify the vehicle's expected behaviour during the mission (e.g. the depth of the water column, the strength and direction of underwater currents, the battery endurance for vehicles operating in $-10°C$ temperatures), then they are more likely to choose larger (more constrictive) safety margin strategies.



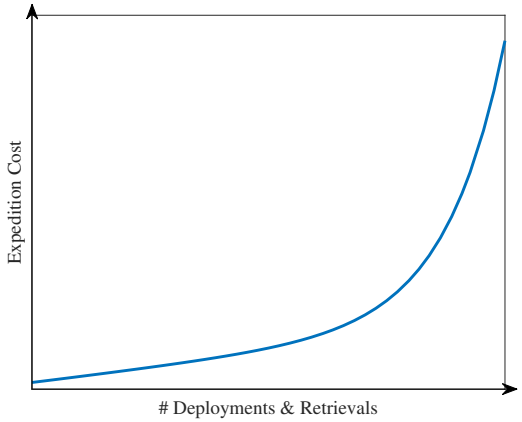
(a) Risk to vehicle versus operator specified safety margins.



(b) Safety margins versus operator confidence in predicted mission progress.



(c) Number of deployments/retrievals versus risk to vehicle.



(d) Cost versus number of deployments/retrievals in AMV expeditions.

Figure 1.1: Set of relationships between cost, deployments, risk, safety margins, and operator confidence that show that properly quantifying the behaviour of AMVs in the environment increases cost-effectiveness.

Failure analysis and risk investigations particularly amongst the Autonomous Underwater Vehicle (AUV) and glider community (Stokey et al., 1999, Brito et al., 2014, Brito and Griffiths, 2016) has shown that a disproportionate amount of failures occur during deployments and retrievals of the AMV. This is partly due to the fact that many faults are immediately noticeable once the vehicle is in the water or while warming up, but there is also the fact that humans launching and retrieving vehicles at sea is a difficult and risky business that demands a lot from the crew. AUVs can be dragged and banged up the sides of ships, gliders are sometimes hit by the tendercraft sent to retrieve them, bulkhead seals can be forgotten or incomplete, and communications masts may be broken by retrieval gaffs. These, and many other incidents (Stokey et al., 1999), are good indicators for the rule of thumb: *"The more that humans touch*

the vehicle, the more likely that something will break". Ideally, operators would like to minimise the number of deployments and retrievals that have to be done, and this requires a very thorough understanding of the risks to the vehicle (as in figure 1.1c).

This thesis considers the quantification of risk to the vehicle during the execution phase (i.e. post-deployment/pre-retrieval) of the mission. During the execution phase, the risk of something failing onboard the vehicle increases with mission duration (Brito et al., 2014). Exposing the vehicle to increased risk will of course increase the likelihood of making more deployments and retrievals, but also not exposing the vehicle to "enough" risk means the vehicle is performing smaller, quicker missions, resulting in more deployments and retrievals over the entirety of the expedition.

Following this line of reasoning, the cost of an expedition involving AMVs is likely to increase non-linearly with the number of deployments and retrievals simply because there is an elevated risk of failure occurring during these phases, resulting in additional resources spent in recovering from the failure (figure 1.1d). The links between each of the relationships above reveal that there is a clear cost benefit for finding better ways of modelling the marine vehicle planning domain (i.e. quantifying the behaviour of the vehicle, and the effects of the environment on the vehicle). From a mission planning perspective, better quantification of the planning domain means a higher confidence in the mission plan, which leads to an optimised set of safety margins, an acceptable amount of risk, a lower amount of deployment/retrieval operations and, ultimately, dollars saved.

Recently, constraint-based temporal planning has seen success in generating feasible plans for AMVs. Actions are modelled in terms of time duration and then are scheduled to minimise the total time duration of the mission. Operators introduce safety margins into the planning process by providing the planner with the maximum and minimum expected durations for each action. However, this method does nothing to truly quantify the behaviour and response of the vehicle, relying on the operator's experience from past missions.

A dynamic model of the vehicle can be used to simulate the predicted motion obtained from performing an action, thereby obtaining the duration with less uncertainty. Simulation comes with a higher computational resource cost for performing simulations of action queries. The simulation model must also reliably represent the marine environment. Any poorly measured or unmeasured components will contribute to deviation between the simulated and actual model. If the model deviates outside of the expected bounds, the vehicle may become stranded from

overcommitting its resources to the mission, or become inefficient from undercommitting.

From the deployed vehicle's perspective, an essential state to monitor is the Point of Safe Return (PSR): the point at which the vehicle must return to a rendezvous point in order to be recovered. For this state to be estimated, the battery supply's state-of-charge must be estimated through measurement of the battery voltage and current discharge. An inaccurate model of the battery's state-of-charge can lead to a potential under- or overestimate of the PSR, resulting in premature rendezvous or in stranding respectively. More problems with the marine environment are discussed in chapter 2.

The vehicle spends energy doing work against the environmental and hydrodynamic loads imposed on it (external loads), as well as spends energy running onboard systems (such as navigation sensors, payload instrumentation and communication modules, referred to as the hotel load). Additionally, the vehicle's power supply can be represented as the number of Joules (J), or Watt-hours (Wh) it is capable of supplying. Representing the vehicle's planning resource as energy instead of time accounts for not only the motion of the vehicle, but also the expected loadings on the vehicle. Furthermore, the energy consumption of the vehicle is capable of being calculated from the measured voltage and current draw of the battery supply. These two qualities make a compelling case for investigating energy as a planning resource for AMVs.

1.2 Aim and Research Questions

The aim of this thesis is: *to develop, implement and demonstrate a new automated planner that increases the survivability and reliability of a deployed fleet of AMVs.*

In pursuit of the aim, the following questions need to be answered:

1. What are the present challenges facing the state-of-the-art in mission planning for AMVs?
2. How can energy consumption be used as a planning resource?
3. Can an energy-aware mission planner improve the reliability and/or the survivability of vehicles?
4. What is necessary to integrate an energy-aware mission planner into a real world AMV deployment?

1.3 Thesis Structure

This thesis is structured around chapter 2, chapter 3 and chapter 4, which are reproductions of accepted or submitted entries in peer-reviewed journals and provide answers to the research questions in section 1.2. The final sections of chapters 2 and 3 (2.6.2 - 2.6.4 and 3.4.3 - 3.4.6) are follow-on discussions of the work presented in each chapter, and link the respective content with the following chapter within the context of the thesis aim.

Chapter 2 identifies the state-of-the-art in mission planning for marine vehicles. The survey is diverse, and is intermingled with literature from other categories of robot planning as well as optimisation and operational research. There are many challenges that only confront the marine vehicle planning domain, which are also identified in chapter 2. The chapter further discusses these challenges, and introduces the concept of planning according to the energy capacity and consumption of the deployed vehicles. Chapter 3 formally defines the energy planning domain, proposes an automated planner method for producing vehicle plans within this domain, and concludes with testing of the planner on a set of simulated data sets for an offshore wind-farm inspection mission. The final component of chapter 3 discusses the limitations of only using simulated data to develop a planner for marine vehicles, and attempts to identify some of the key elements of a real vehicle platform that cannot be incorporated directly into the energy planning domain. Chapter 4 presents a prototype autonomous surface vehicle platform that is to have the energy-aware planner deployed on. Major modifications to the planner are made to enable it to consider certainty of current energy consumption measurements with respect to the planning prediction. The new planner and vehicle platform were tested in a lake environment, and the data relevant to describing performance of the decision-making of the vehicle is presented. Chapter 5 concludes the thesis by presenting the major outcomes from each chapter, and discussing the opportunities for future work that these outcomes present.

1.4 Novel Components

Key challenges to automated mission planning for marine vehicles as well as a road-map of critical research paths that will address these challenges. This contribution (chapter 2) assists future research in the field with a signposted reference.

The automated mission planner proposed (chapter 3) uses energy as a planning resource (found in Thompson and Galeazzi (2018)), which is a novel contribution to the marine vehicle planning field. Using energy as a planning resource diverges from the state-of-the-art planners, which

operate in the temporal domain. The energy domain is more complex to model, but has a fuller represenation of the vehicle's interactions with the environment than temporal modelling.

This thesis also contributes to the Operations Research (OR) field with a method for reducing the planning problem domain through a spatial clustering heuristic, and the adaptation of an existing meta-heuristic optimisation solver to the marine vehicle planning problem. In chapter 4, further modifications to the planner enable it to be deployed in the real world. The final modified planner is unique in that it plans with a more complex domain model that incorporates uncertainty through survival functions, allowing the operator to modify the vehicle's behaviours based on risk.

Finally, a LSTM neural network was proposed as a data-driven method for learning to forecast the energy consumption of a vehicle based on previous mission history. This final component addresses the energy modelling problems that stem from incorrectly identifying the dynamics of the vehicle by replacing it with a black-box model. This approach has not been implemented on marine vehicles before, and could impact several research areas that rely upon dynamic models including: automated planning, model predictive control and simulation.

References

Bermudez, L., Graybeal, J., and Arko, R. A. (2006). A marine platforms ontology : Experiences and lessons. In *Proceedings of Workshop on Semantic Sensor Networks*, Athens, GA, USA.

Brito, M. and Griffiths, G. (2016). Autonomy: Risk assessment. In Curtin, T., editor, *Springer Handbook of Ocean Engineering*, pages 527–544. Springer International Publishing, New York, NY, USA.

Brito, M., Smeed, D., and Griffiths, G. (2014). Underwater glider reliability and implications for survey design. *Journal of Atmospheric and Oceanic Technology*, 31(12):2858–2870.

Chrissley, T., Yang, M., Maloy, C., and Mason, A. (2017). Design of a marine debris removal system. In *2017 Systems and Information Engineering Design Symposium (SIEDS)*, pages 10–15, Charlottesville, Virginia, USA.

Ferguson, J., Pope, A., Butler, B., and Verrall, R. (1999). Theseus AUV - two record breaking missions. *Sea Technology*, 40:65–70.

Kahneman, D. and Tversky, A. (1977). Intuitive prediction: Biases and corrective procedures. Technical report, Defense Advanced Research Projects Agency. N00014-76-C-0074.

Kaminski, C., Crees, T., Ferguson, J., Forrest, A., Williams, J., Hopkin, D., and Heard, G. (2010). 12 days under ice - an historic AUV deployment in the Canadian High Arctic. In *2010 IEEE/OES Autonomous Underwater Vehicles*, pages 1–11, Monterey, CA, USA.

Karimanzira, D., Jacobi, M., Pfuetzenreuter, T., Rauschenbach, T., Eichhorn, M., Taubert, R., and Ament, C. (2014). First testing of an AUV mission planning and guidance system for water quality monitoring and fish behavior observation in net cage fish farming. *Information Processing in Agriculture*, 1(2):131 – 140.

Li, X., Bilbau, S., Martín-Wanton, T., Bastos, J., and Rodriguez, J. (2017). SWARMs ontology: A common information model for the cooperation of underwater robots. *Sensors.*, 17(3):569.

Martins, A., Dias, A., Almeida, J., Ferreira, H., Almeida, C., Amaral, G., Machado, D., Sousa, J., Pereira, P., Matos, A., Lobo, V., and Silva, E. (2013). Field experiments for marine casualty detection with autonomous surface vehicles. In *2013 OCEANS San Diego*, pages 1–5, San Diego, CA, USA.

Nguyen, B., Hopkin, D., and Yip, H. (2008). Autonomous underwater vehicles A transformation in mine counter-measure operations. *Defense & Security Analysis*, 24(3):247–266.

Stokey, R., Austin, T., Von Alt, C., Purcell, M., Goldsborough, R., Forrester, N., and Allen, B. (1999). AUV bloopers or why Murphy must have been an optimist: A practical look at achieving mission level reliability in an autonomous underwater vehicle. In *International Symposium on Unmanned Untethered Submersible Technology*, pages 32–40.

Thompson, F. and Galeazzi, R. (2018). Robust mission planning for autonomous marine vehicle fleets. *Manuscript submitted to: Engineering Applications of Artificial Intelligence*.

Chapter 2

Review of Mission Planning for Autonomous Marine Vehicle Fleets[1]

2.1 Introduction

AMV fleets promise significant benefits to end-users and operators of solo AMVs, including longer mission durations, larger scale missions, more complex mission tasks, and a higher likelihood of success in fulfilling the mission objectives. AMVs are any robotic platform that has been configured to move and operate independently in a marine environment. Industry standard platforms include AUVs, Autonomous or Unmanned Surface Vehicles (ASV/USVs), Unmanned Aerial Vehicles (UAVs), and buoyancy-driven gliders.

The deployment and recovery of one AMV is non-trivial, and for a multi-vehicle fleet the challenge is compounded. Each vehicle currently requires a specialised team to plan, launch, and recover it. The scope and complexity of viable missions for organisations are significantly limited by this arrangement. Consider the following case study: The search for the missing Malaysia Airlines flight MH370 lasted three years and involved the deployment of 21 aircraft, 19 ships and 2 AUVs as a joint effort by 8 nations (Geoscience Australia, 2017). The search was suspended in January 2017, even though the likelihood of finding the missing flight within the newly drafted search areas was high (Gilbert and Reinfrank, 2016). The search generated 710,000 square kilometres of bathymetric data, making the Southern Indian Ocean one of the most comprehensively surveyed areas in the world (Geoscience Australia, 2017). AUVs

[1]Sections 2.1-2.6.1 are a reproduction of the article published to the Journal of Field Robotics accepted 13th September, 2018.

played a critical role in bathymetric mapping, side scan sonar mapping, and high definition photography of sites of interest during the search. This data gives significant scientific insight into deep ocean geological features and other oceanographic phenomena (Picard et al., 2017). However, the mission ended without finding the aircraft, costing the nations involved AUD 200 million in total (Geoscience Australia, 2017).

The data produced by the AUVs is set for full release mid-2018, but one question has been left unanswered: *What more can be done to locate the missing aircraft?* The recent "no find, no fee" agreement between both the Malaysian and Australian governments and AMV fleet operator Ocean Infinity indicates that extended deployments of multiple AMVs is the newest and most likely attempt at successfully locating MH370 (Harvey, 2017). Ocean Infinity, having deployed the offshore support vessel Seabed Constructor along with eight AUVs and eight ASVs in the new search area since January 22nd, 2018, managed to cover 4,500 km^2 in the first six days (Ocean Infinity, 2018). This is an exponential increase in survey performance when compared to the initial search made by the solo Bluefin-21 AUV, operated by Phoenix International, which achieved its nominated 860 km^2 search area within 70 days of operation (LeHardy and Moore, 2014).

The demand for autonomous systems capable of completing costly, expansive and dynamic missions has risen over recent years. The search for MH370, environmental observation of large areas of oceans (Meredith et al., 2013), long term maintenance of submerged structures (Christensen et al., 2015), and even in extraplanetary marine environment exploration (Kunz et al., 2009) are a few examples demonstrating a broad cross-section of the scope of these missions. As a researcher it is tempting to become focussed on smaller aspects of the mission requirements and lose sight of the bigger picture. Identifying key research goals that will benefit not just one sector of autonomous marine technology but several, requires detailed investigation of the field.

The drivers behind the deployment of AMVs come from scientific research, industrial applications, and military defence missions. Mapping beneath ice shelves (McPhail et al., 2009, Graham et al., 2013), cetacean monitoring (Bennett et al., 2015), internal and external inspection of vessels, (Bonnin-Pascual et al., 2012, Hover et al., 2012) and underwater vehicle detection for defence purposes (Caiti et al., 2013) are just a few examples of the significant utility of AMVs.

AMVs have most frequently been deployed on solo missions, and the bulk of research and

development surrounding autonomous vehicles has focussed on the solo AMV (see Yuh et al. (2011) for a detailed cross-section of such research projects). We have defined two significant limitations on the impact of solo marine vehicles: mission scope and mission dependency.

Mission scope is defined as the extent of objectives and tasks the vehicle must achieve for mission success. Solo AMVs are limited by the capabilities of the platform, no single vehicle can carry a payload for multi-faceted missions. During the search for MH370, high-altitude bathymetry mapping was performed using towed-arrays from manned vessels, providing seafloor topography data for the mission planning on the Bluefin-21 AUV that is operated at low-altitude for high resolution mapping. The AUV could have completed both the high altitude and low altitude sub-missions, but its payload would have had to be changed (e.g. data offload, battery recharge etc.) in series for each task, doubling the mission time and thereby increasing the activity of the costliest vehicle to operate, the manned vessel.

Mission dependency considers the extent to which mission success relies on the autonomous components of the operation. If the objective of the mission rests primarily on the reliable operation of a solo AMV, large opportunity costs must be accepted. If something goes wrong with the vehicle the mission must be put on hold until the vehicle is returned to an operational capacity. As illustrated by Stokey et al. (1999), faults in a mission are not just vehicle related, most occur at the human-vehicle interface. A need for formal risk assessment of AMVs is critical for the expansion of autonomous capabilities, however no standard practice exists (Brito and Griffiths, 2016b).

Mission dependency and mission scope limitations imposed by solo AMVs may be addressed by introducing AMV fleets. The mission dependency on a vehicle is reduced when there are other vehicles with similar capabilities available for substitution in the event of the primary vehicle experiencing a critical fault. The scope of the mission is also dramatically improved in two root areas by using multiple vehicles. The first area relates to the space-time domain of the mission scope. Because of the increase in the expected performance (such as the Ocean Infinity/Bluefin-21 comparison made previously), larger areas can be covered in smaller time frames. The second area increases the complexity of the mission scope. A fleet can, through a wider range of capabilities and actions, perform tasks that depend upon others. Xiao et al. (2017) provide a good example of this type of mission scope enhancement by using a UAV to identify rescue targets for an ASV specialised for lifesaving capabilities to navigate towards. In this case, the UAV was well suited for the search task and the ASV is capable of performing the rescue task but the ASV relies on successful completion of the search task to be useful.

Further discussion on mission complexity is done in section 2.4.2.

However, the implementation of AMV fleets in missions where the operator's knowledge of the environment is highly limited presents a modelling conundrum for system designers. Designing better navigation systems, state observer algorithms, and obstacle avoidance behaviours enable vehicles to minimise potential fault scenarios in unknown environments, but they do not assist the vehicle with handling faults within these environments. For example, typical fail-safe recourse actions that work well in open sea (such as emergency surfacing) result in an unrecoverable vehicle in under-ice environments, or in potential collision risks with infrastructure or vessels in coastal environments, or in security risks such as detection by the enemy in military missions. The major connection between these scenarios is that significant risk assessment and contingency planning is required on behalf of the vehicle operator and end user to ensure that, given a fault while operating in a limited knowledge environment, the vehicle fails gracefully (and preferably without completely jeopardising future missions). In the case of multiple AMVs, this planning phase is even more crucial because cross-asset risks and contingencies also need to be considered.

The autonomous marine robotics field first adopted mission planning that required little deliberation from the vehicle. AMVs were programmed with scripted, sequential waypoints, with an operator-designed Finite State Machine (FSM) fault handling mechanism that allow the vehicle to abort or jump to predefined positions in the mission script according to the operator's expectations for the mission. FSM vehicles can execute predictable behaviours in response to expected faults or mission progression triggers (see Elkins et al. (2010) for a FSM implementation for multi-ASV patrol operations). Scripted FSM behaviour mission plans are limited to the expectations of the operator and can be inflexible when confronted with unexpected observations. If the operator is out of contact with the vehicle, they will have no way of providing a repair plan for the vehicle, increasing the risk of vehicle failure or a mission abort.

Since 2007, with the implementation of the Teleo-Reactive Executive (T-REX) planner (McGann et al., 2007) on AUVs at the Monterey Bay Aquarium Research Institute (MBARI), AMVs have begun shifting in planning paradigm from scripted FSM mission plans to goal-based deliberative planning, allowing the vehicle to develop its own plan to achieve operator specified goals. AUVs have benefited the most from onboard deliberation due to the communication limitations of the underwater environment preventing operators from providing repairs to the mission plan. In fleet deployments, the planning space is scaled up in complexity as each vehicle contributes its own action-state space to the mission. The T-REX planner does not perform efficiently on

larger scale planning problems (Ingrand and Ghallab, 2017) and has been supplemented by a centralised, shore-side, mixed-initiative planner (Py et al., 2016) in order to formulate goals for the individual T-REX planners onboard each vehicle.

The major requirement for automated deliberative planning is that the platform has been made sufficiently robust, both in terms of physical design and adaptive fault tolerant control, that the planner does not need to be immediately aware of the fault condition of the vehicle. For example, suppose a slow leak is detected onboard an AUV. Automated ballast/attitude control might compensate to a certain extent so that the planner does not need to urgently trigger an emergency return to the surface.

This review seeks to fill in and discuss gaps in knowledge that are specific to AMV fleet mission planning, drawing on literature from marine robotics, planning and AI, multi-robot systems, and human-robot operations research. Naturally, literature surveys exist for each of these specific areas. Yuh et al. (2011) surveyed marine robotics applications, which have advanced considerably since 2011. More recently, Shukla and Karki (2015) surveyed applications of AMVs in the offshore energy industry, with small mention of AMV fleet applications for oil spill detection and recovery missions. Verfuss et al. (2016) surveyed marine robotics for applications in marine sampling and addressed the benefits of AMV fleets, but did not compare planning paradigms. Ingrand and Ghallab (2017) presented a well-distilled survey of the enormous amount of literature addressing deliberative planning for autonomous robots with small discussion on deliberation for multi-robot systems, the take home messages of which we apply specifically to the AMV fleet domain. Finally, Harris et al. (2016) reviewed the risk and reliability models used for multi-AMV operations, the results of which we bring to the discussion of automated planning. To the extent of the authors' knowledge, there are no surveys that have comprehensively covered planning for AMV fleets.

Figure 2.1 summarises in broad strokes the extent of this review. Section 2.2 discusses the intersecting priorities of the major parties that consider AMV planning and deployments in terms of reliability, utility, survivability and quality. Section 2.3 furthers the discussion on AMV fleets and introduces some existing projects that are explicitly developing or using AMV fleets, the majority of which are heterogeneous. Section 2.4 presents the modelling structure for multi-robot planning, referred to as taxonomies within multi-robot systems literature. Section 2.5 presents the challanges to mission planning for AMVs in two primary categories:challenges that the marine environment poses to AMVs (2.5.1), and the challenges of implementing taxonomies from 2.4 into an AMV fleet (2.5.2). Sections 2.5.3 and 2.5.4 present literature that are meet-

ing some of these challenges, whilst fulfilling the priorities of the major parties discussed in section 2.2.

Section 2.5.5 identifies and discusses future challenges for AMV fleet research, identified from the discussions in the previous sections. These key challenges, if addressed, will greatly increase the capabilities of AMV fleets and foster interest in development of emerging autonomous marine systems technology. For researchers specialising in multi-robot systems, multi-agent planning, or intelligent systems, solutions to these challenges may already exist and need a little support to be brought to the autonomous marine systems field. Briefly, the challenges are:

1. Automated decomposition of abstract objectives into sequences of machine executable primitives.
2. Automated prediction of survivability for a vehicle, given an environment and a set of tasks.
3. Estimation of a vehicle's reliability of performing a task correctly, allowing for a probabilistic estimation of success for the mission plan.
4. Learning to adapt mission parameters based on historical performance of the fleet.
5. Incorporation of morphological and self-reconfigurable robots into existing planning domain paradigms.

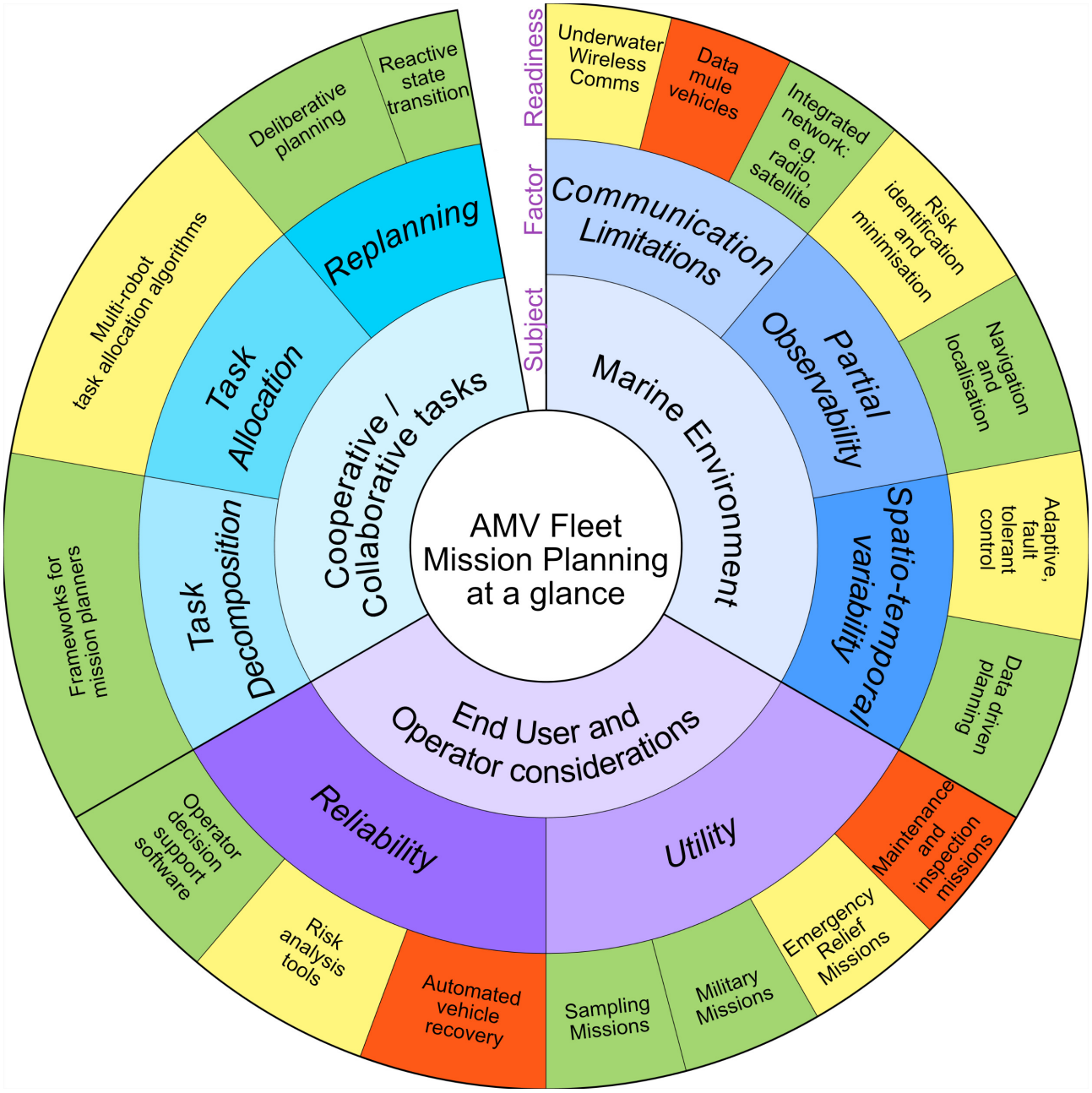


Figure 2.1: Summary of the major topics covered in this review. The inner ring classifies the major categories (or subjects) of challenges and considerations for AMV fleets, the middle ring decomposes these challenges into more specific areas (factors). The outer ring represents key contributions that address the challenges and considerations, and the 'readiness' of these contributions are indicated by red (still in the lab or proof-of-concept field tests), yellow (multiple field tests but not yet a standard), and green (implemented consistently in the field and maturing).

Finally, the breadth of this survey locates the readiness of mission planning for multi-vehicle marine robotics systems on several dimensions. The primary recommendations are summarised as follows:

- Multi-vehicle deployments have attractive advantages over solo vehicle deployments but require a degree of automated mission planning to function well.
- Deliberative planning is the current state-of-the-art for adaptive tasking of marine vehicles and is replacing fixed mission script plans.
- Disruption tolerant communication networks and deliberative planners are enabling large-scale, complex deployments.
- Complexity of deployments is limited because operators need to monitor the operation, and sanity check the decisions being made by deliberative planners.
- Planners either focus on deliberation or task allocation, the operator performs the neglected component.
- Future fleet planners should incorporate both deliberation and task allocation to allow the operators to focus on monitoring and risk assessment.

2.2 End User Specifications: Survivability, Reliability, Quality, and Utility

End users purchase or commission the production of AMVs to fulfil niche requirements within their interests. Their relationship to an AMV system can be best characterised (in this case for a science mission) with a use case diagram in the Unified Modelling Language (Object Management Group, 2017) in figure 2.2.

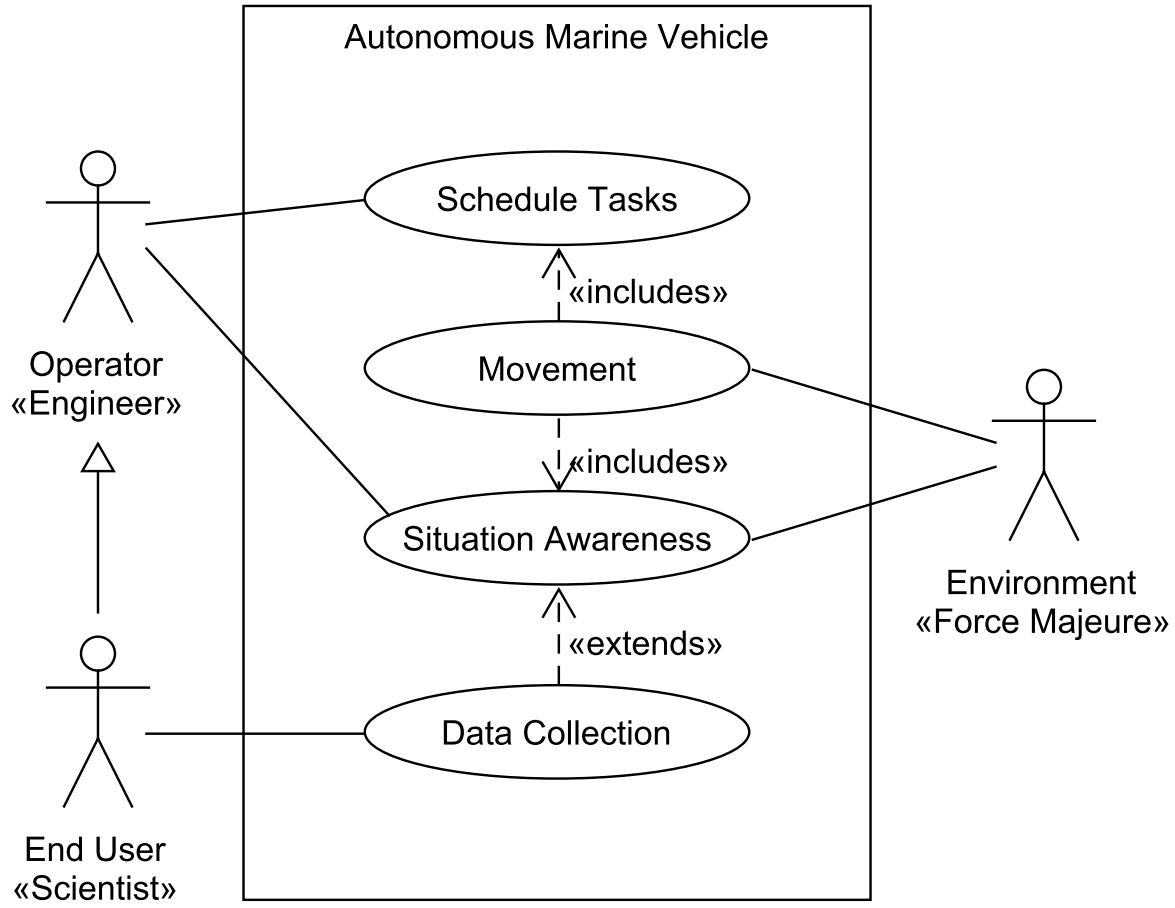


Figure 2.2: General use case of AMV system for scientific mission. The base ability of an AMV is to sense the environment (Situation Awareness), and to actuate movement through the environment. The end-user (a scientist) is usually only interested in a subset of the environment that the vehicle is sensing. The vehicle's movement depends on both the instructions given to it by the operator, and navigational/obstacle data sensed from the environment.

Brief descriptions of the main actors interacting with the system are as follows:

End User: Provides the abstract mission objectives of the AMV to the operator and receives the major outcomes (in this case scientific data) from the AMV mission. In some situations, the end user may also be the operator.

Operator: Schedules tasks for the AMV through a mission programming language, deploys and recovers the AMV, and monitors the AMV for mission progress and updates.

Environment: Provides constraints and loadings on the movement of the AMV, and provides sensory data that is interpreted by the situation awareness behaviour of the AMV.

During the acquisition, preparation, and deployment of their platforms, end users and operators have an interest in the following performance characteristics of the AMV:

Survivability: Of most importance to the operator is that the vehicle survives the mission regardless of the mission outcomes. For the operator this may be just to recover mission log data to diagnose faults for future mission planning. The operator does not want to write off the cost of an AMV lost at sea, nor do they want to spend similar or more resources in recovering the vehicle. For the end user the will for survivability is primarily to fulfil mission objectives, such as the recovery of captured data.

Reliability: Of secondary importance to the operator is that the vehicle completes its mission objectives as reliably as possible. This component requires an estimate of the likelihood of failure that considers the vehicle profile, mission profile, and environment profile.

Quality: Aside from fulfilling mission objectives, the end user wants the outcomes of the mission to be as favourable as possible to their specified measures of performance. For example, scientists want high confidence in their collected data with accurate position and timestamp tags.

Utility: Ultimately, end users want the vehicle to complete missions successfully, within their specified scope and allocated budget. On the other hand, operators continue developing the capabilities and versatility of their vehicles and supporting infrastructure so that they can attract a larger market base of end users and sustain ongoing business. Sometimes the end user and operator must work together on feasibility and mission scope to develop vehicles with the right capabilities, and within budget. Kaminski et al. (2010) illustrates an example of such partnerships and the sizeable resource costs required to develop agreeable and actionable missions.

2.3 Autonomous Marine Vehicle Fleets

Proponents of Multi-Robot Systems (MRS) have argued that fault tolerant design can be supported by deploying multiple vehicles that have redundancy based robustness (Kalwa, 2009). Missions that are considered too risky for a robust, fault-tolerant solo vehicle might instead be considered acceptable using a robust, fault-tolerant and redundant fleet. Mission times are also reduced by dividing objectives up amongst a multi-vehicle team (Fu et al., 2013). Additionally, fleets can tackle significantly more complex missions than solo vehicles (Aguiar et al., 2009). A cross-section of research projects, other organisation initiatives, and field reports that focus on AMV fleets is given in table 2.1.

As a starting point, the projects can be grouped into four categories: environment sampling, military, emergency relief, and inspection and maintenance. The level of autonomy of the mission planning for each category was also found to decrease in this order, where environment sampling has examples of sophisticated deliberative autonomous planners. This might be in part due to the passive nature of environment sampling (i.e. the vehicles only sense and move through the environment, they do not actively endeavour to change it), which simplifies the action-effect relationships that planners must search through to formulate viable plans. Additionally, marine environmental monitoring is by far the largest sector and has the largest deployment size of autonomous vehicles for a single mission (see Wynn (2016) and the Marine Autonomous Systems in Support of Marine Observations (MASSMO) project for examples of such deployments). Within MASSMO, the vehicles themselves have their own individual schedules and sampling paths to follow, and do not interact with each other in meaningful ways. However, each mission yielded significant quantities of environmental, acoustic, and bathymetry data.

Table 2.1: Literature in AMV fleets grouped by fleet composition.

Fleet Composition	**Category**	**Project Title**	**Notable Publications**
ASV, glider	Sampling	Marine Autonomous Systems in Support of Marine Observations	Wynn (2016)
AUV, glider	Sampling	Autonomous Ocean Sampling Network	Bellingham and Zhang (2005)
	Sampling	Autonomous Ocean Sampling Network II	Fiorelli et al. (2006)
Heterogeneous ASVs	Military	Autonomous Maritime Navigation Project	Elkins et al. (2010)
	Sampling	Sensor Coverage with Heterogeneous ASV Fleet	Forooshani and Jenkin (2015)
Homogeneous AUVs	Military	Underwater Acoustic Network Field Report	Caiti et al. (2013)
	Sampling	Range Formation Control of AUVs	Soares et al. (2013)

Fleet Composition	Category	Project Title	Notable Publications
Heterogeneous AUVs	Military	Cooperative Mine Countermeasures	Sotzing and Lane (2010)
	Maintenance	Self-reconfigurable, Modular, Collaborative Underwater Robots	Christensen et al. (2015)
ASV, AUV	Maintenance	Smart and networking underWAter Robots in cooperation Meshes	Real-Arce et al. (2016)
ASV, UAV	Emergency	Emergency Response Field Reports	Murphy et al. (2008), Xiao et al. (2017)
	Sampling	Remote Sample Collection and Whale Monitoring	Bennett et al. (2015)
AUV, UAV	Sampling	Mixed-Initiative Oceanographic Field Experiments	Faria et al. (2014),Py et al. (2016)
AUV, UAV, ROV	Emergency	Easter Japan Tsunami Recovery Field Report	Murphy et al. (2012)
AUV, passive float, glider	Sampling	Southern Ocean Observing System	Meredith et al. (2013)
ASV, AUV, UAV	Sampling	Sunfish Tracking Using Aerial, Surface, and Underwater Vehicles	Pinto et al. (2013)
	Sampling	Network of Heterogeneous Autonomous Vehicles for Marine Research	Ludvigsen et al. (2016)

Sampling missions also feature the most advanced mission planners currently implemented in AMV fleets. MBARI, in collaboration with the University of Porto (Pinto et al., 2013) used AUVs, ASVs and UAVs to track tagged sunfish using the Laboratory of Underwater Systems and Technology (LSTS) toolchain (Ferreira et al., 2017) developed for networked marine vehicles. This toolchain has been further tested in collaboration with MBARI, the Norwegian University of Science and Technology (NTNU), and LSTS in Faria et al. (2014), where AUVs and UAVs simultaneously sampled oceanographic data above and below the surface. It was found that limited underwater communication made it difficult for operators to follow the

behaviour of the vehicles in real-time, who were concerned about the automated decisions being made by the onboard deliberative planner. Further field experiments in seabed surveying were completed using AUV, UAV and ASV resources and the LSTS toolchain by Ludvigsen et al. (2016). It was concluded that the network produced effective coordination between aerial, surface and submerged resources, but required direct communication with the operator to achieve this result. In the latest trials of the LSTS toolchain, a mixed-initiative planner called EUROPtus (Py et al., 2016) was implemented on a fleet of AUVs and UAVs to perform cetacean monitoring (similar to the mission objectives in Bennett et al. (2015)). The centralised, deliberative planner can decompose abstract operator tasks and schedule them to nominated vehicles, allowing the operator to focus on risk analysis and sampling strategies for each individual vehicle.

Naval missions introduce extra complexities and constraints to the AMV fleet that aren't present in passive sampling missions. Missions often require the vehicles to patrol or sweep an area for targets, which requires them to have sufficient situation awareness for target identification. Furthermore, the scope of the mission may require the vehicles to avoid detection or operate within obstacle filled areas (such as minefields). The work by Ocean Systems Laboratory and spin-off company, SeeByte, has produced a mission planner that coordinates a heterogeneous fleet of AUVs for mine countermeasure missions (Sotzing and Lane, 2010). The mission planner uses Hierarchical Task Networks (HTNs) for task decomposition (where the hierarchy is constructed by the operator) and a simple task allocation strategy that allocates outstanding tasks to the nearest idle vehicle that has the required payload (e.g. intervention AUVs are selected for disarming tasks over torpedo AUVs). Communication between vehicles is done via periodic acoustic broadcasting, allowing mission updates from each vehicle to synchronise with the rest.

Surface AMV fleets benefit from the increased bandwidth of radio communication, and can adopt a centralised command and control architecture. Elkins et al. (2010) present the Autonomous Maritime Navigation project, which featured ASVs capable of automatically identifying and locating targets (such as unknown vessels) and sharing this information with a command centre and other ASVs in the area. The vehicles were field tested for solo patrol and sentry missions, and one cooperative sentry mission. Mission behaviours were programmed as FSMs, where state transitions were triggered by target identification, geographic location, or after time-outs. This type of reactive behaviour programming allows the system to be easily debugged (which was ideal for field testing) but puts the entirety of planning deliberation onto

the operator. Caiti et al. (2013) presented another example of finite state reactive behaviour planning for AUV fleet acoustic monitoring missions.

AMVs are also being used for search and rescue and emergency relief missions. These missions are difficult for *a priori* mission planning because the environments the vehicles are deployed in are situational and unique, meaning that the operators must develop mission plans *in situ*. Field report literature published by Robin Murphy and the Center for Robot-Assisted Search and Rescue show that in most cases to accomplish meaningful relief work, the vehicles must be controlled with remote pilots rather than autonomous agents. Murphy et al. (2008) made use of a piloted UAV to localise a piloted USV that was operating in a Global Positioning System (GPS) denied environment. Similarly, Xiao et al. (2017) used a piloted UAV to identify targets from a birds-eye vantage point, which was used as waypoint information for a rescue task enabled ASV.

Compared to the other categories, field-based literature on maintenance, inspection, and other commercial missions for AMV fleets is in short supply, despite forecasts for significant AMV growth in the offshore energy market (Westwood, 2016). This is thought to be in part due to the risks associated with using existing autonomous vehicle technology being too high for safety-driven industries such as offshore oil and gas (Shukla and Karki, 2015). Currently, AMVs are predominantly being used in solo deployments for survey and inspection missions. Hastie et al. (2018) proposed to improve the transparency of AMV decision making for offshore energy missions, so that the operator can reliably predict the effects of vehicles participating in intervention missions. Real-Arce et al. (2016) present the Smart and networking underWAter Robots in cooperation Meshes (SWARMs) project, which aims to provide the architecture to control collaborative AUVs and ASVs for offshore structure maintenance and inspection missions.

2.4 Collaboration or Cooperation: Nuances in Multi-robot Systems Taxonomy

The MRS research field has defined several taxonomies to help categorise the problem space and develop algorithms specialised for individual problem types. Within the problem domain of mission planning, the authors have identified three primary problem subsets: *task decomposition*, *task allocation*, and *mission replanning*.

2.4.1 Task Decomposition

Mission planning for robots is not just a case of calculating where a vehicle should be navigating to (Brooks, 1987). The planner, whether it is a human operator or an AI, must understand the mission requirements, and must have a method of decomposing these abstract requirements into a set of atomic, or primitive tasks (Korsah et al., 2013) that can be allocated to vehicle resources. As formalised in Russell and Norvig (2003), abstract tasks are valid if they can be represented in the highly detailed vehicle spaces without contradiction or conflict. The primitive tasks must be compiled and then executed by the vehicle resources.

Decomposition Strategies: The process of breaking an abstract objective into a sequence of actions that can be executed by a machine is situated within the deliberative mission planning phase (Pettersson, 1997). Planners are characterised by their representation of the state-space, and the method used to search for a valid solution that navigates from an initial position to a goal within the state-space. The full spectrum of planners developed in AI literature is incredibly varied, meriting several literature surveys on several subsets. In the case of AMV fleets, only a small subset of the available literature is relevant. Table 2.2 provides a categorised snapshot of planner paradigms that address critical components of planning, some of which have been applied to AMVs. Readers are also directed to Ingrand and Ghallab (2017), who provide an extensive review of deliberative planners for autonomous robot systems.

The Stanford Research Institution Problem Solver (STRIPS) first sought to decompose tasks using a formal logic relationship language between goal fulfilment and action effects (Fikes and Nilsson, 1971). Many STRIPS-like planners now exist, such as the Goal Oriented Action Planner (Orkin, 2004), which has seen success in discrete (primitive action) decision-making AI for games such as F.E.A.R.. In efforts to formalise STRIPS, the Planning Domain Definition Language (PDDL) was developed as a standardised language as part of the International Planning Competition (Ghallab et al., 1998). PDDL Planners allow an agent to generate a set of available actions to manipulate the initial state of the agent to some specified goal state, provided that the operator has given detail to a sufficient number of available actions. The limitation of these planners is that they assume a quiescent environment, a description rarely applied to the marine space. In the case of marine vehicles, many of the platforms do not have the capability to stop and wait for a new set of observations, meaning that the planner must also consider temporal availability to perform actions.

In the special case of AUV fleets, Giger (2010) developed a high-level mission planning language

using lexical parsing to automatically generate AUV mission task code, based on user-made human-readable specification code. This procedure allows the operator to quickly generate scripted mission code for the vehicles, but limits vehicles' deliberative autonomy to scripted fault handling.

Planning that considers reaching specified goals in a finite time and with finite resources is trying to address what is known in the literature as the temporal constraint problem. Barták et al. (2014) presents a detailed introduction to the mechanics of temporal planning. Several extensions to PDDL have been made to incorporate temporal logic schema, such as the New Domain Definition Language (NDDL) and PDDL2.1 and 3.0. Many temporal planners exist in the literature, most of which have been developed and implemented on spacecraft and extra-terrestrial robots, such as Indexed Time Tables (IxTeT) (Ghallab and Laruelle, 1994), Mixed-Initiative Activity Plan Generation (MAPGEN) (Ai-Chang et al., 2004), and EUROPA2 (Sara Bernardini, 2007).

In AMV fleets, the EUROPtus planner (Py et al., 2016), based on Extensible Universal Remote Operations Planning Architecture (EUROPA), for multi-vehicle scheduling has been implemented in conjunction with the T-REX planner (McGann et al., 2007), an onboard temporal planner originally developed for AUVs. At the operator level, the EUROPtus planner finds optimal task schedules for separate AMVs based on the goals developed by the operator for each vehicle. The T-REX agent onboard the vehicles receives this schedule and further decomposes the tasks through its modular (or reactor) design. The major challenge with T-REX is in resolving scheduling conflicts when there are competing reactors on the same hierarchical level, requiring careful design choices for prioritisation. EUROPtus ensures that the priority for goals is determined on the operator side, mitigating this issue. A genetic algorithm approach by Miloradović et al. (2017) is designed to allocate and schedule abstract tasks for a fleet of heterogeneous AUVs, but has yet to be implemented on simulated or real robots.

Temporal planners deal indirectly with the uncertainty of the effects of actions (indeterminism) by specifying duration bounds for scheduled tasks (i.e. a shortest possible duration and the longest expected duration). These duration bounds then allow the planner to search through and schedule tasks on a global timeline which operators can interpret easily. The duration bounds are determined by the operator, who use their expert knowledge and historical data to produce reasonable estimates.

Planners that directly consider non-determinism have been modelled as Partially Observable

Markov Decision Processess (POMDPs). POMDP planners are advertised as a 'universal planner', able to map any state to an appropriate action using an acquired policy. However, the complexity of the search space for policy acquisition is $\left(2^k\right)^n$ for n k-ary states, making POMDPs hard to implement for large state domains (Ingrand and Ghallab, 2017). Furthermore, POMDPs are learned using data driven techniques such as reinforcement learning, which makes them difficult to be implemented *in situ* by operators. Several methods exist in the literature for simplifying the planning-acting space for the POMDP framework, most of which use heuristic search methods from classical planning for areas of the space that are considered "sufficiently determinant". Readers interested in using POMDPs for planning are directed to Mausam and Kolobov (2012) and Spaan (2012).

To aid in task decomposition, planners can be represented as hierarchical translations between abstract actions mapped to increasingly primitive actions. Hierarchical architectures, especially HTNs, have been the most commonly applied planners in robotics (Ingrand and Ghallab, 2017). The hierarchical language allows for the operator to easily specify the relationship between abstract tasks and primitive actions but relies upon the expert knowledge of the operator to provide the hierarchy. For AMV fleets, Lesire et al. (2016) developed a HTN planner that searches for a deterministic plan prior to deployment. Each vehicle then executes its own local plan and uses an operator-specified failure-remedy hierarchy to repair the plan during execution. Sotzing and Lane (2010) used a HTN structure as a template for users to perform their own task decomposition during mission planning. Both planners were implemented on heterogeneous vehicles performing mine countermeasure missions.

The last consideration for planners of AMV fleets is that they, on some level, plan for multiple acting agents. It is important to clarify that the agents referenced to in this section are individual vehicles, rather than hardware or software components. Planners up to this point have provided operator-friendly representations to allow the operator to decide individual goals for the vehicles. Multi-agent planning attempts to automate this component by taking global goals of the mission and then decomposing them into the planning spaces of the individual agents. A more comprehensive state of the multi-agent planning (hardware or software) literature is available in Torreño et al. (2017). Multi-agent planning for AMV fleets has been in the literature since Sotzing et al. (2007), where the agents represented individual AMVs. The previously mentioned genetic algorithm approach by Miloradović et al. (2017) represents a multi-agent system of AUVs using PDDL for temporal domain planning. The key differences in multi-agent planners are discussed further in section 2.4.4.

Table 2.2: State-Action planning methods (with a focus on multi-agent planners) categorised by domain complexity. Domains that allow for abstract-primitive decomposition, consider temporal constraints such as time-windows and limited availability, consider partial observability of the current state and/or the predicted effect of an action (indeterminism), and are structured for multiple agents are more complex but also more closely represent the AMV fleet domain. Dots indicate the domains that each planning method was developed for.

	Deterministic		**Multi-Agent**		**Temporal**	**Partially Observable**
	Decomposition	*Primitives*	*Co-operative*	*Negotiated*		
STRIPS-based PDDL-based		●			●	
MA-STRIPS MA-PDDL		●	●			
Hierarchical Task Networks	●				●	
Dec-POMDP		●		●		●
MacDec-POMDP	●			●	●	●

Multi-agent planning that considers partially observable states and non-determinist actions has been implemented for robot planning as a Decentralised POMDP (Dec-POMDP). The Dec-POMDP framework presented by Oliehoek and Amato (2016) seeks to distribute the modelling of the POMDP across multiple agents, resulting in policies developed for each agent that cooperatively contribute to the fulfilment of the mission goal. A further extension to Dec-POMDP is to allow for decomposition of abstract tasks down to primitive actions, which has been realised using the theory of options (Sutton et al., 1999) to create a Macro-Action Decentralised POMDP (MacDec-POMDP). Liu et al. (2017) used reinforcement learning with a state-of-the-art iterative sampling based maximum expectation algorithm to learn a MacDec-POMDP model for the planning and acting of a heterogeneous fleet of ground and aerial vehicles performing a search and rescue mission.

2.4.2 Multi-robot Task Allocation

As discussed in the previous section, T-REX and EUROPtus allow the operator to quickly decompose and schedule tasks which has been put to great use in sampling missions, but the planning is done on a vehicle-by-vehicle basis, leaving the allocation of tasks to the operators. Another planning method for AMV fleets considers the task allocation problem for multiple agents (a key component of multi-agent/multi-robot planning). The allocation of a set of machine readable tasks to a set of vehicles is commonly known as the multi-robot task allocation problem. Gerkey and Matarić (2004) identified a taxonomy to describe classes of problems and corresponding solutions for multi-robot task assignment.

Single Task vs. Multiple Task Robots (ST vs. MT): In most situations, a vehicle is only capable of undertaking one task at a time. However, in certain situations such as multi-sensor data gathering, vehicles can execute multiple tasks at the same time.

Single Robot vs. Multiple Robot Tasks (SR vs. MR): Some complex tasks require several vehicles working on it at the same time. The task can be decomposed into an ordered set of primitive tasks that must be completed with overlapping time windows and use resources from several robots.

Korsah et al. (2013) furthered the description of ST-MR and MT-MR problems by providing definitions for compound task interrelations and dependencies.

Instant Assignment vs. Time Extended Assignment (IA vs. TA): Instant assignment means that a newly identified task can be allocated to a fleet node straight away. Time extended

assignment simply means that tasks must be scheduled to be allocated in the future. In the context of marine based missions, TA problems are more realistic, as communication lag, asynchronous tasks, and vehicle travel times are considered.

The definition for TA problems was expanded to include time window constraints (Nunes et al., 2017). In this case a task must be completed in a specific time interval. Synchronisation of task timing and precedence constraints, where a task cannot be started until another specified task is complete are also considered.

The concepts of utility and, to a certain extent, quality (discussed in section 2.2) are represented in multi-agent allocation problems by the utility function, defined by Gerkey and Matarić (2004) as:

$$U_{rt} = \begin{cases} Q_{rt} - C_{rt} \\ -\infty \end{cases} \tag{2.1}$$

Where Q_{rt} and C_{rt} are the quality and cost estimate of a robot r completing a task t. This is a model for an objective function and has been solved for AMV fleets, with varying degrees of success (discussed further in section 2.5.2).

Additionally, the utility of one vehicle can be modified by other vehicles' behaviours. The relationship between each vehicle's utility can be used to classify this dynamic as altruistic-cooperative, rational-cooperative, egoistic-cooperative, or uncooperative (Düring and Pascheka, 2014). Ideally, rational-cooperative behaviours should be prioritised for achieving mission goals.

In the case of AMV fleets, various methods have been used to allocate tasks to vehicles. Sotzing (2009) used a simple heuristic that selected vehicles for a new task based on the nearest idle vehicle with the required payload. Communication was minimised by having each vehicle perform the heuristic onboard using an internal knowledge database of the other vehicle's locations and statuses. Sotzing accepted the small risk that vehicles could be allocated the same task, trusting in the update protocol of the vehicles to resolve the conflict *in situ* (i.e. at some point one vehicle will be identified as closer than the competitors). Sariel (2007) used an auction-based method to allow AUVs to bid for segments of a search area for a mine countermeasure mission, however this was only validated in a Monte-Carlo simulation. Giger (2010) developed and compared integer-linear programming, greedy, and multi-objective genetic algorithm approaches to allocate waypoints to a homogeneous fleet of AUVs. The genetic algorithm was capable of waypoint allocation that minimised the time taken to survey a target area whilst minimising the total number of vehicles required.

2.4.3 Mission Replanning

A mission may have to be replanned during execution because significant changes have been detected in the vehicle systems (such as faults), the environment, or by user input. Upon detection of a major change to the system, the mission planner (automated or operator) must first detect and classify impact of the change so that the likely effects to vehicles and the existing plan can be estimated and predicted. Finally a new plan may be developed based on these estimated causal links.

An AMV is able to observe some of its surroundings with sensors that must be processed into an estimation of its current state. The components of the state that are of importance to the completion of the plan depend upon the planning domain. As a simple example, Galceran et al. (2015) developed a terrain-following trajectory planner for an AUV that actively adjusts the planned trajectory based on sonar measurements. The planner is strictly a motion planning algorithm (Kalakrishnan et al., 2011) that finds a new trajectory based on a 3D occupancy map generated by a sonar Simultaneous Localisation and Mapping (SLAM) package. In this case, change detection and classification is handled at the sensory perception level (i.e. detecting new occupancy levels for the 3D map), and the motion planner replans by performing its function periodically.

A similar procedure occurs in mission planners that work at the more abstract level of goals, tasks and actions. Temporal planners such as Neptus (Pereira et al., 2006) and EUROPtus (Py et al., 2016) periodically obtain estimates of the time remaining for a vehicle to complete a task (based on its kinematic state information). When the time error between updates causes the estimated task completion time to slide outside of the expected completion window, or when a task is added or deleted by an operator: a full temporal planning step is performed based on the most current state information. As Rajan et al. (2013) points out, frequent updates to the planner are required to minimise the total amount of time when the AMV systems are operating on plans that are out-of-date.

Detection of changes in the environment are not as well represented in the temporal domain. For example, a planner that generated a plan based on a certain weather condition will replan with poorly estimated time windows if it is not made aware of changes in sea-state, wind speed or currents. Models that estimate and predict weather conditions based on recent measurements (Hollinger et al., 2016) will assist the planner in assessing environmental effects if an appropriate environment-vehicle interaction model is available (see chapter 8 of Fossen (2011) for examples

of current, wind and wave force models for marine vehicles).

2.4.4 Fleet Architecture

The makeup of the fleet structure is analogous to terrestrial and aerial MRS architectures. The major characteristics of the architecture are described as follows:

Communication network structure: The communication network is a key component to the responsiveness of the fleet to updates to the mission structure, but faces implementation challenges that have been best summarised by Akyildiz et al. (2004). Broadcasting communication can ensure as many nodes as possible are updated across the network, at the expense potential corrupted transmissions due to multiple vehicles broadcasting at once and the potential wastage of bandwidth transmitting redundant information. Unicast communication conserves bandwidth and allows for targeted communication but requires that the sender and receiver must be within range of each other. A selective form of broadcasting based on location known as geocasting allows for local broadcasts within a geographical region (Chen and Pompili, 2010).

Homogeneous vs. Heterogeneous Composition: Two major fleet compositions exist in MRS research (Farinelli et al., 2004) and both have been applied in AMV fleets (see section 2.3 for examples). Homogeneous fleets consist of a node network of clone vehicles. Swarm relative localisation algorithms (Delight et al., 2016), leader-follower algorithms, coordinated path following algorithms (Aguiar and Pascoal, 2007, Xargay et al., 2012, Häusler et al., 2013), and a leapfrogging method (Matsuda et al., 2013) are best suited to the organisation of homogeneous fleets.

Heterogeneous fleets consist of a node network of vehicles that vary in capabilities and operational domains, and constitute most of the projects presented in section 2.3. In practice, a heterogeneous fleet consists of vehicles specialised at performing different tasks (typically due to cost constraints), meaning that the implied task allocation problem domain constraint for these fleets is that no one vehicle can do all of the tasks demanded of the fleet, making the process of task allocation easier by selecting against vehicles that do not have the required payload (e.g. see the allocation method in Sotzing et al. (2007)). However, this is only a special case formulation of the heterogeneous task allocation problem, which is more complex for fleets that are designed with redundancy (i.e. having several vehicles capable of completing the same task, but at different levels of survivability, reliability, utility and quality).

Unlike homogeneous fleets, where the planner is designed to interface with a set of cloned

vehicles, heterogeneous fleets require a software framework that is platform independent so that mission plans can be executed using a common protocol. In AMV fleets, many network interfaces, and command and control architectures have been implemented for specific heterogeneous missions. The MOOS-IvP (Benjamin et al., 2006), CaSHMI (Pastore et al., 2017), Neptus (Pereira et al., 2013), CARACaS and R4SA (Huntsberger and Woodward, 2011), HiDDeN (Lesire et al., 2016), and ROS (Forooshani and Jenkin, 2015) are all implementations that bring the low-level control system specific to the vehicle up to a common interface, each of which have been used for heterogeneous AMV fleet missions.

The current standard in heterogeneous AMV fleet planning is vehicle-by-vehicle task planning using EUROPtus, a centralised, mixed-initiative temporal planner (Py et al., 2016). Mixed-initiative planning is powerful because it allows the operator to quickly describe custom tasks for the planner to schedule and allocate, making it adaptable to planning many kinds of science missions. Due to the centralised architecture, EUROPtus is vulnerable to unreliable communication. NECSAVE (Pinto et al., 2017), takes the temporal planner and distributes it as a common platform interface for any vehicle. Individual vehicle planners synchronise opportunistically in a similar method to the BIIMAPS architecture (Sotzing et al., 2007).

Cooperative vs. Collaborative Missions: This component considers the relationship between nodes as they work on tasks. If the assigned nodes are dividing a set of similar tasks between them, then they are acting with a cooperative behaviour. A good example of this is the parallelised AUV mission planner developed by Giger et al. (2007), which allows a homogeneous AUV fleet to optimally subdivide an area of seabed to survey.

If the assigned nodes have been allocated a set of dependent, but dissimilar tasks, then they are acting with a collaborative behaviour. The heterogeneous fleets deployed in Pinto et al. (2013), Faria et al. (2014) and Py et al. (2016) for marine sampling co-temporally with different sensory payloads on each vehicle are excellent examples. A collaborative mission that involves a more obvious dependency is in the detection and subsequent disarming of mines as addressed by Sotzing and Lane (2010).

The distinction between cooperative and collaborative missions is made because the computational load of parallelising a cooperative mission is significantly less than allocating collaborative behaviours to nodes. Cooperative missions, especially missions that involve parallelisation, do not have to consider as many interrelating dependencies. Homogeneous fleets naturally fit into this parallelised schema, and so it comes without surprise that homogeneous fleets are well

suited to performing cooperative missions. Heterogeneous fleets, being able to perform a larger variety of tasks than homogeneous fleets, can undertake a more complicated, interdependent set of mission objectives than their counterpart. This makes them most suitable for collaborative missions. In general, cooperative missions are attractive to users who wish to maximise their coverage of area and minimise their time. Collaborative missions are attractive to users that require complex, interdependent tasks to be performed.

Direct Effect vs. Support Nodes: In collaborative missions, some nodes may not provide direct effect to the assigned tasks, but instead provide support to other nodes who are directly addressing the tasks. For example Schneider and Schmidt (2010) simulated ASVs that shadow an AUV to function as radio-acoustic communication gateways for the operator base station. Bhadauria et al. (2011) proposed data mules for recovering data from sensor nodes, and Ocean Infinity (2018) uses ASVs to act as communication and localisation supporting vessels for the AUVs performing bathymetry tasks. This supporting vehicle type can be extended to numerous support roles such as stranded vehicle data recovery, recharging stations for aerial and underwater vehicles, and as communication relays for mission updates for distant vehicles. In addition, the option of recovering stranded vehicles can allow flexibility in fault tolerance to favour completion of a mission over survival of the vehicle.

Additionally, self-reconfigurable robots (Roehr et al., 2014, Christensen et al., 2015) enable vehicles to support others by attaching and becoming part of a greater physical platform. This presents a unique problem to MRS: *how can robot nodes that are capable of joining to become singular vehicle platforms be incorporated into MRS planning structure?*

Centralised vs. Decentralised Planning: Centralised planning relies upon a single agent (a vehicle, external planner or a human operator) to allocate tasks to the members of the fleet. Decentralised planning divides the planning problem into non-overlapping problem subsets that can be solved by individual nodes and then compared across the network for the best solution. A radical form of decentralised planning is distributed planning, where individual vehicles all contribute to the formalisation of the plan. Within the multi-robot planning domain (i.e. planning and acting agents are bodily separate from one another and must communicate through an external system such as radio, acoustics, optics, or tethered connection), each of these planning topologies have advantages and disadvantages that are strongly linked to the network architectures that they share names with (van Steen and Tannenbaum, 2017). Each of these planning paradigms can be compared on complexity, communication requirements, and fault tolerance as in table 2.3 below.

Table 2.3: Qualitative comparison of centralised, decentralised, and distributed planning architectures (derived from network comparisons made in (van Steen and Tannenbaum, 2017))

	Centralised	**Decentralised**	**Distributed**
Complexity	Simple. Plan generated on one node only. No conflicts.	Intermediate. Plan generated on a few nodes. Conflict resolution required	Difficult. Plan generated by all nodes. Propagation of information. Conflict resolution required.
Communication requirements	Bandwidth bottlenecked through the central node.	Less severe bottlenecks across several nodes.	Broadcasting can cause interference between nodes.
Fault tolerance	Failure at central node causes general failure.	Conflicts in plan. Section failures, no general failure.	Conflicts in plan. Local failures.
Relevant Work	Pereira et al. (2006)	Py et al. (2016), Lesire et al. (2016)	Zlot (2006), Sotzing (2009)

Centralised planning is attractive to operators because there is only one planning agent that must be monitored. Due to the uncertainties of the marine environment, the planning agent must regularly receive measured mission state updates from the acting agents and send plan updates to the agents. An acting agent that fails to receive updates from the planner may be in danger of failure or becoming incapable of completing mission objectives. If the planner has out of date state information, then it can provide dangerous or useless plans to the acting agents. For a centralised planning schema to work effectively, communication between the acting agents and the planning agent must be reliable and regularly available so that the risk of either of the mentioned planning failures is sufficiently low. The EUROPtus planner (Py et al., 2016) is a centralised planner designed to schedule and allocate abstract tasks for individual vehicles. Each vehicle's onboard local planner further decomposes these tasks into a schedule of vehicle specific behaviours. The central planner is not in continuous communication with the vehicles and relies upon the vehicles to establish communication to update state information and repair vehicle plans that the vehicle's local planner cannot repair.

Decentralised planning has reduced bottlenecking of communication traffic by having multiple planning agents that handle different subcomponents of the full mission plan. The added

benefit is that acting agents are less likely to be out of contact of a planning agent, making them less likely to become irrelevant to the current mission objectives. However, ensuring that the planning agents do not allocate acting agents with conflicting tasks (e.g. double booking an agent with different tasks, allocating the same task to multiple agents, or allocating mutually exclusive tasks to agents) makes decentralised planning more complex than centralised planning.

HTNs are suitable for decentralised planning, Lesire et al. (2016)'s HiDDeN planner uses the HTN structure to initialise a mission that is distributed across multiple vehicles. If an independent task is failed, the vehicle involved can locally perform a repair with its knowledge of the overall mission. If the task has higher order tasks depending on it that are being executed by other vehicles, HiDDeN has an implemented synchronisation behaviour to ensure a plan repair is executed only across the vehicles that need to be repaired. Unaffected components of the mission are preserved, and the only communication cost is between the affected vehicles.

Distributed planning does not require reliable communication between all vehicles to create actionable plans but has the highest risk of creating conflicting plans and has inefficient procedures for updating the overall mission plan. This is analogous to distributed systems, whose nodes rely on flooding or random walk searches to obtain useful data from other nodes (van Steen and Tannenbaum, 2017). The BIIMAPS model by Sotzing et al. (2007) eliminated task conflicts by implementing a blackboard broadcasting system for all vehicles, ensuring that updates from specific fleet members incrementally propagate out to the rest of the fleet. Vehicles simply allocate themselves to tasks that are available for allocation on the blackboard according to their feasibility and proximity to the task. If another vehicle has also allocated itself the same task, then the conflict is detected when one of the vehicles receives a broadcast from the other vehicle. The vehicle that is furthest away drops the task, removing the conflict. BIIMAPS does not scale as well as HiDDeN because large fleets of broadcasting vehicles will interfere with each other's messages.

2.5 Challenges in AMV Mission Planning

2.5.1 Marine Environment

The marine environment presents many challenges to both AMV and AMV fleet research. Some of these challenges appear within the literature as terms whose meaning may be opaque to readers from outside of these fields. We have defined these challenges as follows:

Sparse fleet distribution: Fleet nodes are far away from each other. This can result in significant adaptive planning delays due to communication lag, asynchronous updates, and extended travel times to new targets.

Inaccessibility: Some fleet nodes are unable to access other fleet node domain spaces. AUVs in under-ice operations are inaccessible to surface or aerial vehicles.

The next three challenges address aspects of the same root cause: the partially or completely unknown nature of the marine environment (especially in the underwater domain).

Unstructured environment: The environment is difficult to navigate because the layout is not completely known, and objects of interest, such as obstacles and objective targets are not static or completely known *a priori*. This makes it difficult to generate a reliable path for the vehicle to follow, meaning that obstacle avoidance and onboard path planning is crucial for the vehicle. Additionally, the environment may not have enough structure for feature-based localisation, such as SLAM, to build a reliable map.

Dynamic environment: Wind and wave loadings, ocean currents, and thermal currents present additional loading considerations to model-based navigation systems. Black box systems (such as artificial neural networks) are particularly challenged if the loadings were not included in training data (see section 4.5.2 for one such example). This also affects the vehicle at the control level. If the expected response of the vehicle to a control input is not robust enough, the vehicle will have difficulty keeping to planned trajectories. This directly affects the reliability of the mission plan, which is in danger of underestimating travel time or energy consumption of the vehicle.

Spatio-temporal variability: Due to the dynamic environment characteristic, data sets (including bathymetry/occupancy) captured by AMVs will change over time, and often it is the rate of change that is of the most interest to the end users. Adaptive sampling and autonomous science strategies that focus on mission planning centred around gathering interesting or significant data are discussed further in section 2.5.4.

Examples of dynamic phenomena include tide induced sea-levels, density (especially in estuarine environments), free-floating ice, vessels and man-made structures, flotsam, sea-bottom debris, complex ecosystems such as coral reefs, fauna, and suspended particulates. For vehicles that build a local map for SLAM based navigation, old maps may not be valid because the stored features do not match up with the newly sensed environment.

Communication Limitations: Influenced by the above challenges, communication between fleet nodes suffers from bandwidth restrictions, particularly in underwater communication which use acoustic modem transmissions. The recent development of underwater optical wireless technology allows for high bandwidth connections between vehicles at distances of up to 200 m. Kaushal and Kaddoum (2016) present a comprehensive survey of optical wireless communication as well as hybrid acoustic-optical communication architectures. Such optical wireless technology is capable of transmitting video feeds (Woods Hole Oceanographic Institute, 2017), which will in the future increase the operator's knowledge of the environment without necessitating a ROV operation.

Regulatory Limitations: Without regulatory consensus or even adaptable examples, AMV operators create their own methods for determining liability in AMV missions (Manley, 2007).

Fleet nodes that operate in areas close to humans must observe laws for safety at sea. A notable example is the International Regulations for Prevention of Collisions at Sea (COLREGS), which have been applied to ASVs (Benjamin et al., 2006). As the technologies surrounding AMVs mature, more specific regulations will come into effect and vary from jurisdiction to jurisdiction. The Australian Maritime Safety Authority is currently in the process of producing policies to regulate autonomous vehicles operating in Australian waters (Judson, 2017).

In addition, liability in incidents that involve deliberative planning AI must be clear for every deployment, necessitating that automated planning must be translatable to human readable form at every level of control. This presents a tricky situation for neural network based deliberative planners as the process by which the planner decides on a solution is hidden from the designer. Research into revealing the inner decision process of neural networks has seen success in computer vision (Zeiler and Fergus, 2014) where self-learned features that influence decision making can be easily visualised.

Human Factor Limitations: The marine environment imposes its own loads upon the operators of AMVs. The differences between planning from onshore vs. planning while deployed are primarily physical stresses such as fatigue and sea-sickness. Manley (2007) presents several risks to operators and vehicles, particularly during launch and recovery operations.

2.5.2 MRS Technology

Numerous field reports for autonomous vehicle deployments have commented on the impact of the Human-Machine Interface (HMI) on mission success. In the case of monitoring and

replanning for multiple AMVs over an extended deployment, the argument for automated planning assistance for the operator to reduce cognitive task load is strong.

Field reports from real missions (Kruijff et al., 2012) and competition environments (Finn et al., 2012) make key observations on the effectiveness of the HMI in multi-vehicle missions. Edmondson et al. (2014) have made efforts to simplify the multi-vehicle mission plan visualisation HMI using group automation algorithms implemented through the MADARA KaRL engine (Edmondson and Gokhale, 2011). The use of AI to simplify planning reduced the number of actions the user needed to input to achieve a valid mission plan, when compared to generating the equivalent plan from widely available UAV mission planners such as the 3DR mission planner. By reducing the amount of actions that the human had to take while planning, Edmondson et al. (2014) argue that the risk of the human making an error while planning has been reduced.

With special consideration to AUVs, Nelson et al. (1992) produced several layers of software for the visualisation and control of AUVs at the planning, execution, and debrief phases of a mission. Fast-forward to recent developments, AUV HMIs are able to infer plans of vehicles from their trajectory data and present these inferences in human readable sentences using natural language processing Hastie et al. (2017).

Stokey et al. (1999) identified that in AUV operations a significant portion of mission faults were directly caused by the operator inputting incorrect instructions to the AUV. Operator related faults are compounded when managing multiple vehicles with differing capabilities in a complex fleet mission. With a well-designed HMI, an operator may be able to manage a handful of vehicles at once, but HMIs can only partially mitigate the effects of cognitive load Sweller (1988).

Cognitive task load was represented by Neerincx (2003) in three dimensions: time occupied, task set switches and level of information processing. The results of his investigation reveal that the operator can be in danger of 'cognitive lockup' or 'overload' if the level of information processing, time occupied monitoring the task, and the amount of switching between tasks is high. The non-linear scale of the planning problem for AMV fleets means that, at some point, there will be a limit to the complexity of both missions and fleets. At this point automated mission planning becomes a necessity to simplify the planning process to the extent that a human planner can understand and oversee.

Murphy et al. (2008) produced an estimate for the number of human operators (N_h) required

for the deployment of (N_v) AMVs:

$$N_h = 2 \times N_v + 1 \tag{2.2}$$

This estimate has limitations; the number of humans required also depends upon the environment profile and on the profiles of the individual vehicles, and if the time required to complete the mission lasts for several working shifts. Kaminski et al. (2010) engaged with a large team of experts and operators to work in a hazardous environment with just one AMV.

Increasing the number of operators to distribute cognitive load and implementing robust project management is a valid but weak solution. The potential for miscommunication within the operator hierarchy may lead to poorly defined mission plans for the fleet. The added resource costs spent managing and supporting extra operators also may not be economically viable for all end-users.

In addition, there is a practical limit to the number of human operators that can accompany their assigned vehicles during a single expedition. The only way to overcome this limitation is to increase autonomy of planning, fault handling, and risk prediction and minimisation.

Planning Algorithms

Fleets experience limitations in communication (section 2.5.1), causing infrequent updates to the mission planner from these nodes. A real-time agent prediction method implemented by Sotzing et al. (2007) used a common AUV behaviour database to predict what an out-of-contact vehicle would be doing based on its most recent status update. This system allows for task allocation to be executed on each vehicle independently without necessitating communication between nodes. However, the prediction step does not factor in the potential occurrence of a fault aboard an out-of-contact vehicle. Predictions for such gaps in mission planning come from risk analysis of the past history of the vehicles (Brito and Griffiths, 2016a).

Scalability is a significant challenge to centralised planners, as the task allocation problem can take time to solve for a large set of vehicles, a large set of tasks, or for very complicated MR-TE type tasks where there are many viable solutions that need to be assessed.

Distributed solvers designed to solve the Distributed Constraint Optimisation Problem (DCOP) such as the Swarm-GAP algorithm (Ferreira et al., 2008), LA-DCOP (Scerri et al., 2005) and BIIMAPS (Sotzing et al., 2007) face challenges in communicating the minimum required in-

formation for approximate solution to complex, interdependent tasks. In the case of distributed solvers that use a shared memory such as BIIMAPS and other blackboard (Corkill, 1991) based systems, asymmetrical updates to the shared mission space due to communication blocks can cause several vehicles to be allocated the same task.

2.5.3 Meeting the Challenges of the Marine Environment

Aside from the ever-present challenges for localisation, navigation, and control in marine environments, effective collaborative control of an AMV fleet requires a characterisation of the present and future environment within which it is operating. In the pursuit of efficient data gathering, Heaney et al. (2007) used a non-linear constrained optimisation genetic algorithm approach to model and forecast an ocean environment prior to multi-vehicle deployments. Hollinger et al. (2016) used Gaussian processes to forecast ocean currents from a noisy model for reliable mission path planning. Pereira et al. (2013) provide two examples of using ocean current uncertainty predictive path planning to minimise risk of collision with surface vessels. One example planner took a goal-oriented approach that performed well in low-current speed, low-variability environments. The second planner used a conservative Markov Decision Process (MDP) and performed well in high-risk environments where the current speed is high or highly variable; sometimes loitering in safe areas and waiting for conditions to improve before advancing in its mission.

For collaborative operations in constricting environments, such as in thick ice-layer polar regions, considerations for access to the vehicle by supporting fleet members or manned vessels must also be factored into mission planning.

Communication

As alluded to in multiple sections in this paper, reliable communication with all of the vehicles cannot be assumed. Communication delays and noise are a part of planning for underwater vehicle operations. Several unique solutions have been developed for different applications. The most common method for dealing with out-of-contact vehicles is regularly scheduling rendezvous points in places where communication is reliable. For example, the Slocum gliders used by Schofield et al. (2007) surface to establish communication with the shore operator via satellite phone, enabling remote data retrieval and remote commands to be sent (at higher operating costs associated with using satellite communication systems, and the effort expended by the vehicle to reach the surface).

Acoustic and underwater optical communication have presented the opportunity for submerged vehicles to not have to surface to communicate with each other or the operator, albeit with a low bandwidth or over a short range. As mentioned previously in 2.5.2, the work by Sotzing and Lane (2010) has successfully enabled vehicles to operate within the limited communication environment, albeit with the risk of vehicles being allocated to the same task (which is resolved once each vehicle establishes contact with the other). Toohey et al. (2014) produced a communication method that minimised the bandwidth usage but still allow multiple AUVs to share localisation information with each other. However, for missions that require submerged vehicles to reach a consensus that are within range of each other, noise or multi-pathing phenomena may cause a Byzantine Failure (see Driscoll et al. (2004) for a set of Byzantine Failure case studies in avionics). What is needed are multiple routes and methods for operators and vehicles to communicate with other vehicle nodes, something analogous to self-healing, multi-hopping technologies in wireless communication. Readers are directed to a comprehensive survey on UAV communication networks by Gupta et al. (2015) for such examples. There is ample opportunity for adaptation of above-surface communication topologies for underwater environments. Ferreira et al. (2017) at LSTS have been developing a toolchain that provides the basic infrastructure to network heterogeneous AMVs, using a platform independent architecture and disruption tolerant networking.

2.5.4 Meeting the End User Requirements

Survivability

The key to factoring vehicle survivability into a mission is effective assessment of the risk to the vehicles, and minimising and mitigation of identified risks. A mission plan with a primary focus on survivability will rarely see a vehicle deployed into a situation where the risk of the vehicle becoming stranded or lost is high.

Within the AUV domain, the identification and analysis of risks to AUVs have been a difficult and exemplary field of research since the loss of the Autosub-2 under the Fimbulisen glaciated area (Strutt, 2006). Survivability is typically handled in two stages: planning and fault handling. For AUVs, fault-log analysis and expert judgement from seasoned operators (Griffiths et al., 2009) is the only viable method for risk identification for future missions and is a process that is difficult to automate. However, expert systems-based methods such as a Markov chain state transition approach (Brito and Griffiths, 2011) and a Bayesian networks approach (Brito and Griffiths, 2016b), supplemented with expert knowledge, have been presented as tools for

identifying the high risk stages within AUV missions. These methods have yet to be proven effective.

For an AMV fleet, the complexity of assessing survivability is compounded. Each additional vehicle requires an additional survivability risk assessment, most of which cannot be copied (i.e. a UAV cannot be held to the same risk analysis as an AUV). Potential risks in individual vehicles interacting with each other must also be included in these assessments. Survivability planning for a large AMV fleet may not be possible without automated risk assessment tools to assist.

The second stage is the implementation of a fault handling strategy for the identified risks from stage one onboard the vehicle. This component typically consists of a set of human-made rules that causes the vehicle to deviate from scripted mission behaviour and implement a fault handling behaviour upon the detection of a fault.

Correctly diagnosing a fault onboard a vehicle so that the correct handling strategy can be implemented is the major challenge during this stage. Ernits et al. (2010) and Dearden and Ernits (2013) managed to automate diagnosis of faults in AUVs by using the NASA Livingstone 2 diagnosis engine (Williams et al., 1996).

In the case of AMV fleets, the risk to support vehicles is also a consideration for the fault-affected vehicle. For example, suppose a vehicle tasked with completing certain mission objectives detects an immobilising fault. The vehicle has the option of being recovered by a supporting vehicle, however the recovery operation might be too risky for the supporting vehicle to execute. The fault-affected vehicle must have the capability of assessing whether the risk to involved vehicles is lower than remaining immobilised *in situ* before calling for assistance.

Reliability

Building upon survivability, reliability of AMV fleets factors not only in the physical integrity of the vehicles, but in planning a mission that has achievable objectives with a vehicle capable of adaptively prioritising objectives. Reliability is comprised of many factors, but the authors have identified three categories of factors that make significant contributions to the reliability of a vehicle.

Situation awareness: The vehicle's knowledge of the surrounding environment, the mission objectives, and itself.

The operating environment includes obstacles such as the seabed, other vehicles, animals, and structures, forces such as current, wind, waves, hydrostatic pressure and other hydrodynamic effects, and mission targets. All these elements are observable, and awareness relies on the vehicle's ability to accurately sense and categorise them into an 'environment profile'. Characterising the environment involves primarily the collection of data and the identification of physical phenomena.

On the other hand, the mission profile and vehicle profile are completely up to the operator. The mission profile requires a representation that can be displayed at a human-readable level and decomposed into a machine-readable level. The vehicle profile contains a description of the vehicle's limitations (maximum depth, maximum speed, endurance, and range), kinematic and hydrodynamic details (primarily speed and position), assigned tasks, and payload details (onboard sensor manifest). Finally, situation awareness is key to categorising the state of the vehicle, mission, and environment for performance review during mission debriefing.

Adaptivity: The vehicle's capability to alter the mission plan based on the detection of anomalous changes from the situation awareness component.

To do this, the vehicle must have an expected situation to compare the estimated situation to. Knowledge-based systems allow for humans to automatically generate rulesets that a vehicle can use to form expectations during a mission. Inzartsev et al. (2016) present a set of such techniques for the diagnosis of faults onboard AUVs.

Learning: Data-driven machine learning is contributing to marine vehicle autonomy in perception, navigation, control, and in decision-making. Most of the contributions have been at the sensor processing level of the vehicle, either for perception or localisation purposes, which are outside of the scope of this review. However, reactive based planning architectures that directly map sensor inputs to vehicle actions have benefited from machine learning by mimicking 'curiosity'. Curiosity causes the vehicle to actuate its sensors towards stimulating sources. An example of this comes from Kompella et al. (2017), who implemented reinforcement learning on a humanoid robot to learn associations between moving its arm and manipulating objects it could observe from a camera feed. For AMVs, Girdhar and Dudek (2016) used online topic monitoring on a camera feed to identify low dimensional descriptors, and kept a history of these descriptors to identify "surprising" observations. The vehicle has an increased reliability for capturing useful video data by navigating towards coral, fish, divers and other "surprising" observations in favour of sand and open ocean.

Decision-making for an AMV is influenced by its perception of the current state, the desired final state, and its knowledge model of state transitions. Learning a model of the state transitions allows the planner to take advantage of previously unknown methods for accomplishing a goal. Lane et al. (2015) proposed teaching a hybrid ROV/AUV skills (i.e. a procedure of primitive actions for the onboard deliberative planner to use) by providing a supervised learning algorithm with vehicle data recorded during a piloted demonstration.

At a more abstract level, AMVs have benefitted from the planner learning a representation of risk. Hollinger et al. (2016) used learned Gaussian processes to estimate the uncertainty of predictions made for ocean currents, allowing an MDP risk aware planner to choose safe surfacing locations for glider vehicles. Somers and Hollinger (2016) produced a framework for human-vehicle planning that produced energy efficient, safe trajectories that were learned as a compromise to the end user's ideal trajectory. The vehicle and, by extension, the fleet can, through experience of the end user specified missions, determine parameters for planning trajectories that minimise information regret.

Quality

Quality is an extension of reliability that describes the value of the mission outcomes. Most AMV mission objectives involve the capturing of data, either as an end objective or in service of a higher order task, e.g. identifying and disabling sea-mines. In the case of data collection, quality is represented by the accuracy of the measurements taken, and the accuracy of the vehicle's navigation system.

Quality of mission outcomes is influenced by a broad range of issues, including sensor design, environment perception and estimation, and data fusion, and are beyond the scope of this review. Within the domain of AMV fleets, cooperative adaptive sampling is a relatively new strategy that implements simultaneous data collection at different levels of detail. AMVs that are sampling at a lower resolution can use adaptive sampling algorithms to find points of interest to be sampled at higher resolution by other AMVs. Cooperative data collection can also be implemented across related datasets. For example, some AMVs can collect data sets that infer the presence of another data set (through onboard analysis of the sampled data), which can be flagged for AMVs equipped with payloads designed for the collection of the inferred data (Song et al., 2015).

Utility

As discussed, one of the advantages of AMV fleets is that they can perform several tasks at once, allowing for more complicated missions to be undertaken. Most organisations that have created proof of concept fleet technology are now searching for missions that they can bootstrap their vehicles for. This can become a solution in search of a problem and, without a clear communication channel between end users and designers, will continue to be standard practice. Beneath the surface of these applications, significant challenges to AMV fleet development have been identified and presented below:

Cooperative sensing: For AMV fleets, it is the cooperative applications of sensing and localisation that are at the forefront of MRS research. In particular, Cooperative Simultaneous Localisation and Mapping (C-SLAM) aims to fuse local maps from multiple vehicles into a global map (Saeedi et al., 2016). Making effective use of sensor coverage is also a key consideration for AMV fleets; Sydney and Paley (2014) present a multivehicle trajectory generator for sensor coverage of a data set that varies in time and space. Forooshani and Jenkin (2015) developed a sensor coverage algorithm that arranged model-scale ASVs in a leader-follower formation around a waypoint scripted mothership. Parallelisation strategies have been successfully implemented to divide a large mission into a set of identical survey tasks that can be distributed across a set of vehicles for simultaneous execution (Giger, 2010).

Collaboration: Advanced methods for decomposition from abstract mission goals into machine-readable tasks that can then be allocated and scheduled is required to approach missions where not all tasks are identical and may include interdependencies. Mixed-initiative planners like MAPGEN (Ai-Chang et al., 2004) and EUROPtus (Py et al., 2016) allow engineers and scientists to collaboratively design task schedules for sampling vehicles that maximise utility within time constraints. The result is that the individual vehicles exhibit collaborative behaviours by individually completing their assigned sampling tasks, each contributing to a much larger and complex sampling mission. Interdependencies between tasks are resolved through human deliberation and scheduling.

2.5.5 Key Challenges for AMV Fleets

From this discussion, the authors have identified several key areas in AMV fleet research that, as the field matures, will significantly increase interest from potential end-users of AMVs and foster growth in the AMV fleet sector.

Closed Loop Decomposition: Mission planning converts abstract concepts easily understood by humans (such as mission aim, scope, and objectives) into increasingly discrete and quantified concepts that can be processed by autonomous systems (such as waypoints, set points, and fault conditions). Figure 2.3 summarises the major processes that lie within planning: taking the human-made mission (M), decomposing it into goals (or tasks) (G), decomposing these further into behaviours (B) that should complete the tasks, and these are then allocated to vehicles (V). New goals (N) are identified by the vehicles during execution, and these are fed back into the goal decomposition stage. Automation of the full process is not just a challenge for AMV fleets, but for Multi-Robot Systems in general.

Automated survivability prediction: The complicated task of automating assessment of fault diagnostics and producing predictions of likelihood of vehicle loss for future missions is an extremely valuable tool for end-users.

Automated reliability assessment: Automating the prediction of the likelihood of mission failure requires cause-consequence analysis of previous missions. Once again, the end-user would benefit immensely from the generation of reliable mission plans before any vehicle is put into the water.

Automated learning: Tying in the above challenges in risk assessment, a fleet that can reconfigure planning parameters based on previous mission performance will become an effective tool for long-running projects and deployments. However, automated learning for decision-making must also be transparent to human analysis.

Expanding utility: The deployment of dedicated support type AMVs necessitates an expansion of existing AMV fleet planners to accommodate tasks that do not directly contribute to the mission but would expand the utility of vehicles that are directly completing the mission. Further development into cooperative adaptive sampling for AMV fleets is a good starting point to include supporting vehicles into a mission planner. Advanced autonomous infrastructure, such as docking stations, data hand off points, and other supporting vehicles have also yet to be represented within mission planners.

Reconfigurable vehicles: Reconfigurable vehicles possess the ability to change payload without the launch and recovery operations that must also accompany a typical payload change. Additionally, reconfigurable robots can combine with each other into larger systems, allowing for significant changes to the vehicle profile of the combined system. How this type of combined system should be represented within the AMV fleet planning structure has yet to be addressed

by the existing state-of-the-art planners.

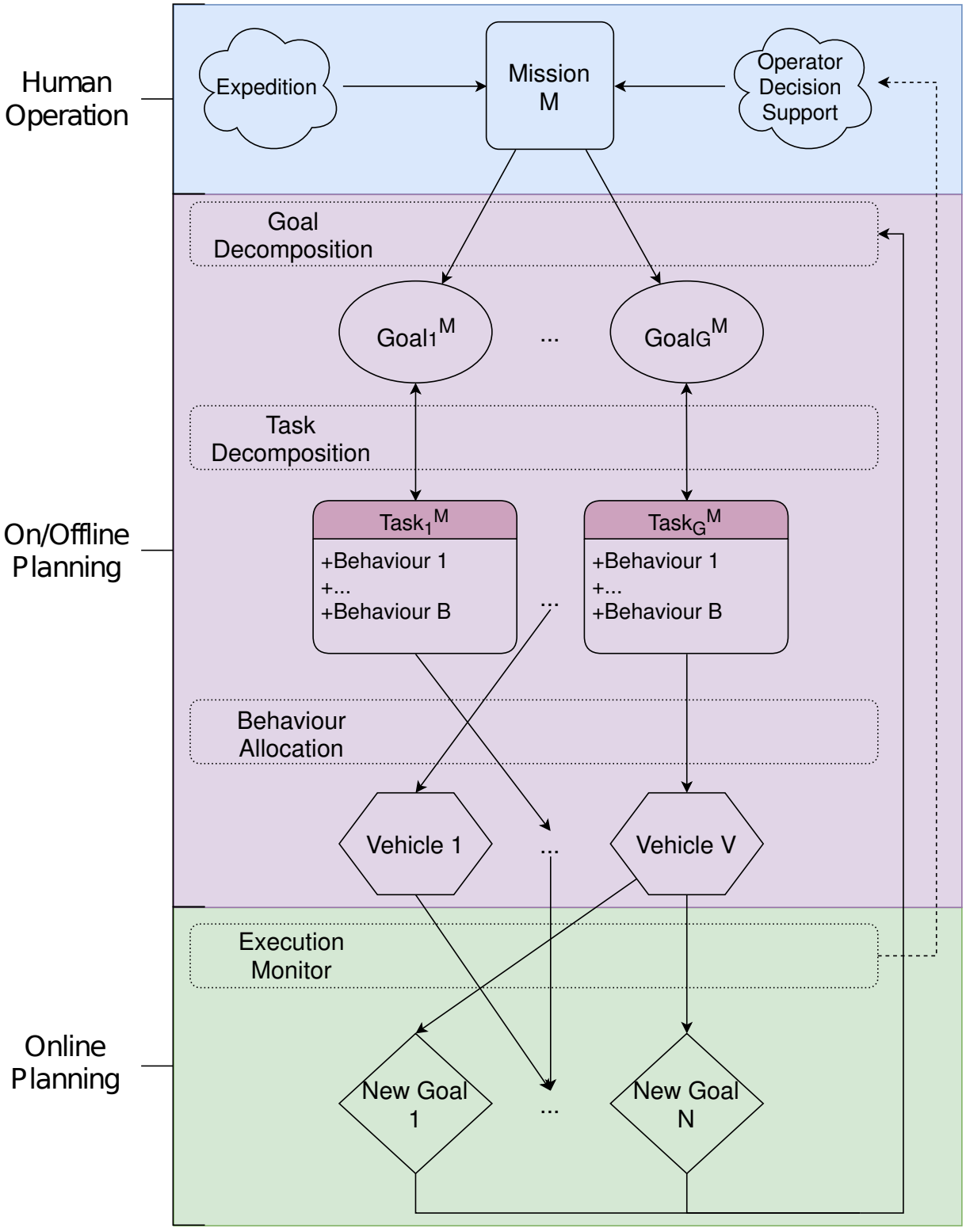


Figure 2.3: Mission decomposition hierarchy. M missions are decomposed into G goals that, when completed, satisfy the mission. Task decomposition is performed on each goal in G, producing a sequence of behaviours that will satisfy each goal. These behaviours are at a primitive enough level to be allocated to the vehicles, abstract tasks can then be decoupled from being entirely dependent on one vehicle. Finally, vehicles report progress in their sequences to the execution monitor, which identifies N new goals for the mission and reports the progress to the operator, who can sanity check the tasks generated.

2.6 Conclusions

2.6.1 Summary

Like most fields of autonomous systems research, AMVs has diversified into many specialised projects that are addressing unique challenges. In developing an AMV fleet system, the designer must be aware of the distinct end user and operator needs for a system that minimises the likelihood of mission failure as a function of both survivability and in reliability. Deploying an AMV fleet puts an exponentially increased burden on the operator to ensure that the planning for all fleet nodes minimises the likelihood of failure. Existing automated planners for AMV fleets have been designed to maximise the utility of the vehicles, most prominently by parallelising the mission into identical sub-missions. The planner, automated or human, must solve a problem that is constrained to meet the intersecting priorities of the end-user and operator across survivability, reliability, quality, and utility. From the meta-perspective of the automated planner's designer, careful consideration, implementation, and testing of automated processes that meet the challenges of the marine environment, end-users and operators is non-trivial.

State of the art planners have begun to incorporate risk-based analysis of the vehicle and the mission to decrease the likelihood of failure. However, migrating these techniques into a fleet arrangement has not been fully considered; each vehicle is individually assessed for likelihood of failure without including vehicle-vehicle interactions as a whole system. The planning architecture for AMV fleets will benefit from being structured around the taxonomies that have been developed in MRS research, with a focus on highly variable and severe environments. In the interests of regulation and liability, automated planners must have decision making processes that are transparent to humans (i.e. in a human readable form). Finally, planners for modular and self-reconfigurable AMVs will have the additional consideration of selecting configurations for the vehicles within a fleet to ensure necessary roles for a mission are fulfilled.

2.6.2 Addressing Reliability in Planning

This chapter presented a review of literature that has influenced the current state of mission planning for marine vehicles. This section discusses the reasoning behind current state-of-the-art mission planners for AMVs, how this reasoning can affect mission reliability and survivability, and provides an introduction to the energy-based planning framework that is proposed in the following chapters. For convenience, mission reliability refers to the capability of the

AMV to successfully complete missions despite uncertainty, and survivability is the AMV's ability to be used for future missions regardless of mission success. Mission planning affects both reliability and survivability, as the plan is the principle decision-making resource used by the AMV while deployed.

In its simplest sense, the most commonly-used function of an AMV is to transport, record and use a payload along a desired trajectory in the marine environment. Developing an appropriate plan requires consideration of the AMV's limitations in performing the trajectory and using a payload. Traditionally, the requirements and limitations are represented as variables that have quantifiable governing resources. For example, the trajectory variable has resource requirements such as the maximum depth and duration of the trajectory. The vehicle variable has resource limitations such as battery runtime, maximum diving depth, and whether a particular payload configuration is present or not. On a higher level of abstraction (but still an equivalent representation of the same process, refer to figure 2.3), the trajectory and payload selection can be represented as behaviours (or actions) that are used to satisfy tasks. The completion of tasks, in turn, is used to fulfil mission goals. For complicated missions where goals and tasks have dependencies or are mutually exclusive, variables to represent and constrain dependencies also need to be considered. A name commonly given to this type of representation is the Resource Constrained Planning and Scheduling Problem (RCPSP). A planner built to solve the RCPSP searches through the variables subject to the constraints of the governing resource metrics to find feasible solutions.

2.6.3 Solving the Resource Constrained Planning and Scheduling Problem

AI Temporal Planners

In the pursuit of generating solutions to the RCPSP, two different fields of research have approached the problem. AI planning researchers model the domain of the problem using a domain definition language, which allows an AI planning agent to search through the problem variables to find a feasible solution. Multiple feasible solutions are usually found, so an AI planner will select the solution with the shortest total duration. The advantage of AI planning is that the flexibility of the domain definition language allows the planning domain to be modelled at a very high level of detail with many layers of abstraction. The disadvantage of AI planning approach is that the solver must use a planning horizon to keep solving times

practical, meaning that the solution may not be feasible past the planning horizon.

Premier examples of AI planners that have seen deployments on AMVs are EUROPtus (Py et al., 2016), T-REX (Rajan et al., 2013) and Neptus (Pereira et al., 2006). Both the EUROPtus and T-REX planners are AMV-centric adaptations of NASA's EUROPA2 temporal-planning framework (Bedrax-Weiss et al., 2004), which stems from the Heurisitc Scheduling Testbed System (HSTS) (Muscettola, 1994) originally developed for the Hubble Space Telescope. EUROPtus creates plans through the constraint-based temporal planning paradigm, which represents resource constraints in terms of time and duration. Goal, task and action variables are all represented by tokens that are subject to these time resource constraints (i.e. A goal must be completed by a certain time). The EUROPA framework can represent goal and task dependencies (i.e. A goal requires one or more tasks be completed in sequence) through a declarative object-oriented modelling language, NDDL (Frank and Jónsson, 2002).

Operations Research

Researchers in the field of Operations Research (OR) formulate the planning problem as an optimisation problem and select or develop an algorithm that searches through the variable space to find a solution. The advantage of the OR planning approach is that efficient algorithms specialised for the formulated optimisation problem can be used to quickly find a good solution. However, the optimisation problem itself runs the risk of not sufficiently representing the planning domain, putting the solver at risk of producing unrealistic plans.

Examples of OR planners for autonomous vehicles include (Evers et al., 2014) and (Tsiogkas and Lane, 2018). In both cases, a variant of the Orienteering Problem (OP) was used. The OP is a bidirectional graph network (figure 2.4) where the nodes of the network represent tasks that can be completed, and the connections (known as edges in graph theory) are the time costs to transition from the completion of one task to the completion of another. To differentiate the OP from the famous Travelling Salesman Problem (TSP) (Hoffman et al., 2013), an additional scalar is assigned to each node to represent a reward for completing a task located at a node. Solving the OP yields a sequence of nodes that will maximise the collected score whilst respecting an overall time limit constraint. Unlike the TSP, solving the OP may not give a solution that visits every node but nonetheless minimises the total travel time whilst maximising the total score. Constraints to the search space can also be enforced by removing forbidden transitions and enforcing the beginning and ending of the solution to "start" and "finish" nodes. Tasks with dependencies can also be encoded into the problem by removing all

transitions to the dependent task except the valid set of supporting tasks.

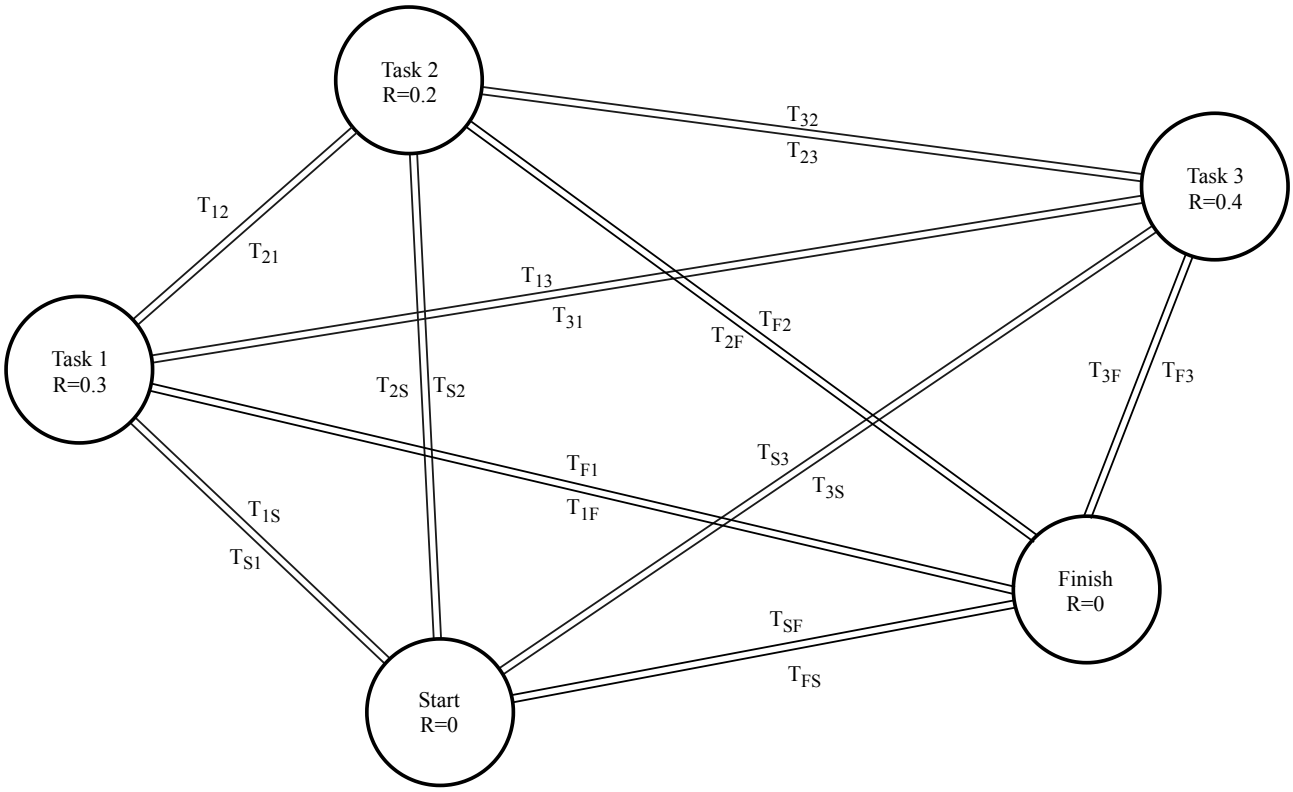


Figure 2.4: Graph representation of the Orienteering Problem's unconstrained search space. Constraints are introduced by pruning edges of the graph according to heuristics. For example, all transitions from Task X→Start and Finish→Task X are forbidden as the starting and finishing points can only be transitioned from and to respectively. Depending on the global time constraint, not every node may be visited.

2.6.4 Problems with Temporal Planning

There are two outstanding advantages to using constraint-based temporal planning (both AI and OR approaches) for AMVs. First, it's simple for operators to interact with and understand. Vehicle action plans are represented as concurrent timelines and can be directly compared with the mission goal and task schedule. Operators can see what vehicle resources are currently assigned through automatically generated Gantt charts, and can see the capabilities of vehicle resources for specific tasks with resource profile charts. Second, it's a pragmatic approach that can quickly generate a plan based on expected time durations. Operators can make changes or add additional constraints to the planner's mission data on the fly without any significant delays from the plan generator.

For a non-military AMV, a large component of its survivability depends upon it being able to maintain power to critical systems (such as an emergency transponder, navigation sensors, and actuators). Developing a plan that does not exhaust the battery resource is extremely

important, so both AI and OR planners model the battery resource with a hard constraint. T-REX and EUROPtus use a resource object to model the battery's state of charge or time remaining as a resource. OR planners usually include the vehicle's endurance as an upper bound within the description of the optimisation problem.

Temporal planners are challenged with providing realistic predictions of time durations for the activities of the vehicle. AI planners rapidly evaluate the expected time cost for a vehicle to complete a proposed trajectory to minimise solving time. The reasoning behind this rapid evaluation is so that the planner can be regularly updated with new information, minimising the effect of prediction errors. OR planners also must evaluate rapidly the expected time cost for task-task transitions, as the full set of allowable edges must be known prior to the solving step. However, resources that depend on the time cost prediction (such as battery state of charge) are susceptible to being inaccurately predicted. This may result in the solver finding plans that are too conservative (the resource constraints are violated earlier than reality) or, even worse, that are too optimistic (the resource constraints are violated later than reality). In both cases, the reliability of the mission plan is undermined.

Finally, most AMV control systems regulate the speed and orientation of the vehicle so that it can follow a reference trajectory. Even though the vehicle is experiencing varying loads, the speed is kept constant which gives a neat linear extrapolation of the time required to complete a trajectory. However, the battery's state of charge may deplete faster on some trajectories than on others, even if the time-costs are equivalent. For example, a vehicle travelling in beam seas will experience higher yawing-moments due to the wind and wave loadings and will have to output more effort to maintain a set speed and orientation than a vehicle travelling in following seas. A temporal planner could select either of these two trajectories because its primary objective is to minimise total mission duration.

Planning with a Higher Order Resource than Time

Planning according to the vehicle's battery state makes optimal use of the resource most critical to its survivability and reliability. It is important to note that battery state planning is sensitive to the environmental loadings, meaning that the planner can make better informed decisions than the temporal planning paradigm. Battery state can be represented in terms of state of charge (%) and endurance (s), but it can also be represented by its energy capacity (either in Watt-hours or Joules). The planning problem can then be formulated through energy methods for estimating the mechanical work required to follow trajectories, and then obtaining the

electrical work required from the battery (subject to its energy constraints). A solver could then find a feasible solution to this energy-based problem that minimises energy consumption while still satisfying task requirements and adhering to the battery capacity constraint.

Battery state is non-trivial to accurately predict because it depends on many variables. The loadings experienced by the vehicle due to the dynamic environment will cause the power consumption to significantly vary according to the controlled actuator outputs. The loadings on the vehicle depend upon the environmental loads (such as wind, waves, and current) which are location and orientation dependent. The vehicle's inertial, hydrostatic, and hydrodynamic loads (which are for the most part non-linear) also depend on the trajectory.

The power consumption is a non-linear function that depends on time, location, orientation, vehicle dynamics and system state, and the battery state. The vehicle's energy consumption is the primary influence on the state of the battery, but other effects such as temperature and pressure, life-cycle, and chemical composition should also be considered[2]. From these conditions, it is impossible to know for certain how many Wh of energy are stored within a battery. In practice, techniques such as counting the number of Coulombs of charge that enter the battery during recharge are themselves uncertain. It is this uncertainty that ensures that operators will always place a minimum charge level safety margin on their vehicles, no matter how efficiently the energy is being used (planning or otherwise).

Monitoring the energy consumption of a vehicle is straight-forward: the voltage across the battery terminals, measured by a voltmeter, and the current flow through the battery, measured by a ammeter placed in series with the rest of the system, will yield the voltage (Volts) and current measurements (Amperes) respectively. A measurement of power (Watts) is then obtained by multiplying the voltage and current. Integration of this power measurement, as will be shown in equation 4.11, yields the energy consumed since measurement was initiated. The voltmeter can be installed as a simple voltage-divider circuit and the ammeter as a Hall effect sensor. The signals outputted by these sensors can be read as analog signals by the onboard computer systems of the vehicle (which typically have analog and digital input/output pins). The measurement noise of the sensors is system dependent, but will cause an error in actual versus measured energy consumption (especially since the noise is integrated over time).

Predicting the energy consumption is less trivial and requires the following:

1. Knowledge of the expected trajectory that a vehicle will take.

[2]These are large fields of research in and of themselves and are outside the scope of this work

2. Estimates for the expected loading that the vehicle will experience along that trajectory.
3. A vehicle dynamics model that obtains the expected actuator states given the loading estimate.
4. A vehicle electrical model that obtains the power consumption of the vehicle given the expected actuator and system states.

The next chapter presents an OR planning approach to AMV mission planning that uses methods for each of the dependencies listed above to estimate the energy consumption of the vehicle for a proposed plan.

References

Aguiar, A. P., Almeida, J., Bayat, M., Cardeira, B., Cunha, R., Häusler, A., Maurya, P., Oliveira, A., Pascoal, A., Pereira, A., Rufino, M., Sebastiao, L., Silvestre, C., and Vanni, F. (2009). Cooperative autonomous marine vehicle motion control in the scope of the EU GREX project: Theory and practice. In *OCEANS 2009 - EUROPE*, pages 1–10, Bremen, Germany.

Aguiar, A. P. and Pascoal, A. M. (2007). Coordinated path-following control for nonlinear systems with logic-based communication. In *2007 46th IEEE Conference on Decision and Control*, pages 1473–1479, New Orleans, LA, USA.

Ai-Chang, M., Bresina, J., Charest, L., Chase, A., Hsu, J. C. J., Jonsson, A., Kanefsky, B., Morris, P., Rajan, K., Yglesias, J., Chafin, B. G., Dias, W. C., and Maldague, P. F. (2004). MAPGEN: mixed-initiative planning and scheduling for the Mars Exploration Rover mission. *IEEE Intelligent Systems*, 19(1):8–12.

Akyildiz, I. F., Pompili, D., and Melodia, T. (2004). Challenges for efficient communication in underwater acoustic sensor networks. *ACM SIGBED Rev. - Special Issue on embedded sensor networks and wireless computing*, 1(2):3–8.

Barták, R., Morris, R. A., and Venable, K. B. (2014). An introduction to constraint-based temporal reasoning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 8(1):1–121.

Bedrax-Weiss, T., Frank, J., Jónsson, A., and McGann, C. (2004). Europa2: Plan database services for planning and scheduling applications. Technical report, NASA Ames Research Center, Moffett Field, CA, USA.

Bellingham, J. and Zhang, Y. (2005). Observing processes that vary in time and space with heterogeneous mobile networks. In *International Workshop on Underwater Robotics for Sustainable Management of Marine Ecosystems and Environmental Monitoring*, Genoa, Italy.

Benjamin, M. R., Leonard, J. J., Curcio, J. A., and Newman, P. M. (2006). A method for protocol-based collision avoidance between autonomous marine surface craft. *Journal of Field Robotics*, 23(5):333–346.

Bennett, A., Barrett, D., Preston, V., Woo, J., Chandra, S., Diggins, D., Chapman, R., Wee, A.,

Wang, Z., Rush, M., and Kerr, I. (2015). Autonomous vehicles for remote sample collection: Enabling marine research. In *OCEANS 2015 - Genova*, pages 1–8, Genova, Italy.

Bhadauria, D., Tekdas, O., and Isler, V. (2011). Robotic data mules for collecting data over sparse sensor fields. *Journal of Field Robotics*, 28(3):388–404.

Bonnin-Pascual, F., Garcia-Fidalgo, E., and Ortiz, A. (2012). Semi-autonomous visual inspection of vessels assisted by an unmanned micro aerial vehicle. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3955–3961, Vilamoura, Portugal.

Brito, M. and Griffiths, G. (2011). A markov chain state transition approach to establishing critical phases for auv reliability. *IEEE Journal of Oceanic Engineering*, 36(1):11.

Brito, M. and Griffiths, G. (2016a). Autonomy: Risk assessment. In Curtin, T., editor, *Springer Handbook of Ocean Engineering*, pages 527–544. Springer International Publishing, New York, NY, USA.

Brito, M. and Griffiths, G. (2016b). A bayesian approach for predicting risk of autonomous underwater vehicle loss during their missions. *Reliability Engineering & System Safety*, 146:55–67.

Brooks, R. A. (1987). Planning is just a way of avoiding figuring out what to do next. Working paper, MIT Artificial Intelligence Laboratory.

Caiti, A., Calabrò, V., Munafò, A., Dini, G., and Lo Duca, A. (2013). Mobile underwater sensor networks for protection and security: Field experience at the UAN11 experiment. *Journal of Field Robotics*, 30(2):237–253.

Chen, B. and Pompili, D. (2010). Reliable geocasting for underwater acoustic sensor networks. In *Proceedings of the Fifth ACM International Workshop on UnderWater Networks*, pages 1–2, Woods Hole, Massachusetts. ACM.

Christensen, D. J., Andersen, J. C., Mogens, M., Furno, L., Galeazzi, R., Hansen, P., and Nielsen, M. (2015). Collective modular underwater robotic system for long-term autonomous operation. In *ICRA 2015*, Seattle, Washington, United States. IEEE Robotics and Automation Society.

Corkill, D. D. (1991). Blackboard systems. *AI Expert*, 6(9):40–47.

Dearden, R. and Ernits, J. (2013). Automated fault diagnosis for an autonomous underwater vehicle. *IEEE Journal of Oceanic Engineering*, 38(3):484–499.

Delight, M., Ramakrishnan, S., Zambrano, T., and MacCready, T. (2016). Developing robotic swarms for ocean surface mapping. In *2016 IEEE International Conference on Robotics & Automation (ICRA)*, pages 5309–5315, Stockholm, Sweden.

Düring, M. and Pascheka, P. (2014). Cooperative decentralized decision making for conflict resolution among autonomous agents. In *2014 IEEE International Symposium on Innovations in Intelligent Systems and Applications (INISTA) Proceedings*, pages 154–161, Alberobello, Italy.

Driscoll, K., Hall, B., Paulitsch, M., Zumsteg, P., and Sivencrona, H. (2004). The real Byzantine generals. In *The 23rd Digital Avionics Systems Conference (IEEE Cat. No.04CH37576)*, volume 2, pages 6.D.4–61–11 Vol.2, Salt Lake City, UT, USA.

Edmondson, J., Cahill, G., and Rowe, A. (2014). On developing user interfaces for piloting unmanned systems. In *International Workshop on Robotic Sensor Networks*.

Edmondson, J. and Gokhale, A. (2011). Design of a scalable reasoning engine for distributed, real-time and embedded systems. In Xiong, H. and Lee, W. B., editors, *Knowledge Science, Engineering and Management: 5th International Conference, KSEM 2011, Irvine, CA, USA, December 12-14, 2011. Proceedings*, pages 221–232. Springer Berlin Heidelberg, Berlin, Heidelberg.

Elkins, L., Sellers, D., and Monach, W. R. (2010). The Autonomous Maritime Navigation (AMN) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles. *Journal of Field Robotics*, 27(6):790–818.

Ernits, J., Dearden, R., Pebody, M., and Guggenheim, J. (2010). Diagnosis of Autosub 6000 using automatically generated software models. In *21st International Workshop on Principles of Diagnosis*, Portland, Oregon. PHM Society.

Evers, L., Dollevoet, T., Barros, A. I., and Monsuur, H. (2014). Robust UAV mission planning. *Annals of Operations Research*, 222(1):293–315.

Faria, M., Pinto, J., Py, F., Fortuna, J., Dias, H., Martins, R., Leira, F., Johansen, T. A., Sousa, J., and Rajan, K. (2014). Coordinating UAVs and AUVs for oceanographic field experiments: Challenges and lessons learned. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6606–6611, Hong Kong, China.

Farinelli, A., Iocchi, L., and Nardi, D. (2004). Multirobot systems: a classification focused on

coordination. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(5):2015–2028.

Ferreira, A. S., Pinto, J., Dias, P., and Sousa, J. B. d. (2017). The LSTS software toolchain for persistent maritime operations applied through vehicular ad-hoc networks. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 609–616, Miami, FL USA.

Ferreira, P. R., Boffo, F. S., and Bazzan, A. L. C. (2008). Using swarm-GAP for distributed task allocation in complex scenarios. In Jamali, N., Scerri, P., and Sugawara, T., editors, *Massively Multi-Agent Technology*, pages 107–121, Berlin, Heidelberg. Springer Berlin Heidelberg.

Fikes, R. E. and Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3):189–208.

Finn, A., Jacoff, A., Del Rose, M., Kania, B., Overholt, J., Silva, U., and Bornstein, J. (2012). Evaluating autonomous ground-robots. *Journal of Field Robotics*, 29(5):689–706.

Fiorelli, E., Leonard, N. E., Bhatta, P., Paley, D. A., Bachmayer, R., and Fratantoni, D. M. (2006). Multi-AUV control and adaptive sampling in Monterey Bay. *IEEE Journal of Oceanic Engineering*, 31(4):935–948.

Forooshani, P. M. and Jenkin, M. (2015). Sensor coverage with a heterogeneous fleet of autonomous surface vessels. In *2015 IEEE International Conference on Information and Automation*, pages 571–576, Lijiang, China.

Fossen, T. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.

Frank, J. and Jónsson, A. (2002). Constraint-based attribute and interval planning. Technical report, NASA Ames Research Center.

Fu, M., Jiao, J., Jianxu, L., and Wang, Y. (2013). Coordinated formation control of nonlinear marine vessels under directed communication topology. In *OCEANS - Bergen, 2013 MTS/IEEE*, pages 1–7, Bergen, Norway.

Galceran, E., Campos, R., Palomeras, N., Ribas, D., Carreras, M., and Ridao, P. (2015). Coverage path planning with real-time replanning and surface reconstruction for inspection of three-dimensional underwater structures using autonomous underwater vehicles. *Journal of Field Robotics*, 32(7):952–983.

Geoscience Australia (2017). The data behind the search for MH370. Geoscience Australia, https://geoscience-au.maps.arcgis.com/apps/Cascade/index.html?appid=038a72439bfa4d28b3dde81cc6ff3214. Accessed: October 2017.

Gerkey, B. P. and Matarić, M. J. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9):939–954.

Ghallab, M., Knoblock, C., Wilkins, D., Barrett, A., Christianson, D., Friedman, M., Kwok, C., Golden, K., Penberthy, S., Smith, D., Sun, Y., and Weld, D. (1998). PDDL - the Planning Domain Definition Language. Technical Report CVC TR-98-003/DCS TR-1165, AIPS-98 Planning Competition Committee, Yale Center for Computational Vision and Control.

Ghallab, M. and Laruelle, H. (1994). Representation and control in IxTeT, a temporal planner. In *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems*, pages 61–67, Chicago, Illinois. AAAI Press.

Giger, G., Kandemir, M., Lovell, S. D., and Dzielski, J. (2007). Automated mission parallelization for a group of UUVs. In *Proceedings of the 15th International Symposium on Unmanned Untethered Submersible Technology*, Durham, New Hampshire.

Giger, G. F. (2010). *An Operator-centric Mission Planning Environment to Reduce Mission Complexity for Heterogeneous Unmanned Systems*. PhD thesis, Pennsylvania State University, University Park, PA, USA. AAI3436143.

Gilbert, E. and Reinfrank, A. (2016). MH370 Experts believe 'high probability' of finding missing plane in new search area. ABC News, https://www.abc.net.au/news/2016-12-20/high-probability-mh370-will-be-found-in-new-search-area/8135404. Accessed: December 2016.

Girdhar, Y. and Dudek, G. (2016). Modeling curiosity in a mobile robot for long-term autonomous exploration and monitoring. *Autonomous Robots*, 40(7):1267–1278.

Graham, A., Dutrieux, P., Vaughan, D., Nitsche, F., Gyllencreutz, R., Greenwood, S., Larter, R., and Jenkins, A. (2013). Seabed corrugations beneath an Antarctic ice shelf revealed by autonomous underwater vehicle survey: Origin and implications for the history of Pine Island Glacier. *Journal of Geophysical Research: Earth Surface*, 118(3):1356–1366.

Griffiths, G., Brito, M., Robbins, R., and Moline, M. (2009). Reliability of two REMUS-100 AUVs based on fault log analysis and elicited expert judgement. In *Proceedings of the*

International Symposium on Unmanned Untethered Submersible Technology (UUST 2009), Durham, New Hampshire. Autonomous Undersea Systems Institute.

Gupta, L., Jain, R., and Vaszkun, G. (2015). *IEEE Communications Surveys and Tutorials*, 18(2):1123–1152.

Harris, C. A., Phillips, A. B., Dopico-Gonzalez, C., and Brito, M. P. (2016). Risk and reliability modelling for multi-vehicle marine domains. In *2016 IEEE/OES Autonomous Underwater Vehicles (AUV)*, pages 286–293, Tokyo, Japan.

Harvey, A. (2017). MH370 Ocean Infinity to be paid millions if new search turns up any trace of missing flight. ABC News https://www.abc.net.au/news/2017-10-30/mh370-search-us-company-promised-millions-for-trace-of-flight/9100818. Accessed: October 2017.

Hastie, H., Liu, X., Petillot, Y., and Patron, P. (2017). Talking autonomous vehicles: Automatic AUV mission analysis in natural language. In *OCEANS 2017 - Aberdeen*, pages 1–5, Aberdeen, UK.

Hastie, H., Lohan, K., Chantler, M., Robb, D. A., Ramamoorthy, S., Petrick, R., Vijayakumar, S., and Lane, D. M. (2018). The ORCA hub: Explainable offshore robotics through intelligent interfaces. In *ACM Human-Robot Interaction Conference: Explainable Robotics Workshop*, Chicago, IL, USA.

Heaney, K. D., Gawarkiewicz, G., Duda, T. F., and Lermusiaux, P. F. J. (2007). Nonlinear optimization of autonomous undersea vehicle sampling strategies for oceanographic data-assimilation. *Journal of Field Robotics*, 24(6):437–448.

Hoffman, K. L., Padberg, M., and Rinaldi, G. (2013). Traveling salesman problem. In Gass, S. I. and Fu, M. C., editors, *Encyclopedia of Operations Research and Management Science*, pages 1573–1578. Springer US, Boston, MA.

Hollinger, G. A., Pereira, A. A., Binney, J., Somers, T., and Sukhatme, G. S. (2016). Learning uncertainty in ocean current predictions for safe and reliable navigation of underwater vehicles. *Journal of Field Robotics*, 33(1):47–66.

Hover, F. S., Eustice, R. M., Kim, A., Englot, B., Johannsson, H., Kaess, M., and Leonard, J. J. (2012). Advanced perception, navigation and planning for autonomous in-water ship hull inspection. *The International Journal of Robotics Research*, 31(12):1445–1464.

Huntsberger, T. and Woodward, G. (2011). Intelligent autonomy for unmanned surface and underwater vehicles. In *OCEANS'11 MTS/IEEE KONA*, Kona, Hawaii, USA.

Häusler, A. J., Saccon, A., Pascoal, A. M., Hauser, J., and Aguiar, A. P. (2013). Cooperative AUV motion planning using terrain information. In *OCEANS - Bergen, 2013 MTS/IEEE*, pages 1–10, Bergen, Norway.

Ingrand, F. and Ghallab, M. (2017). Deliberation for autonomous robots: A survey. *Artificial Intelligence*, 247:10–44.

Inzartsev, A., Pavin, A., Kleschev, A., Gribova, V., and Eliseenko, G. (2016). Application of artificial intelligence techniques for fault diagnostics of autonomous underwater vehicles. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–6, Monterey, CA, USA.

Judson, G. (2017). The regulatory approach to marine autonomous systems. In *Autonomous Underwater Technology Conference*, Perth, WA, Australia. Society for Underwater Technology.

Kalakrishnan, M., Chitta, S., Theodorou, E., Pastor, P., and Schaal, S. (2011). STOMP: Stochastic trajectory optimization for motion planning. In *2011 IEEE International Conference on Robotics and Automation*, pages 4569–4574, Shanghai, China.

Kalwa, J. (2009). The GREX-project: Coordination and control of cooperating heterogeneous unmanned systems in uncertain environments. In *OCEANS 2009-EUROPE*, pages 1–9, Bremen, Germany.

Kaminski, C., Crees, T., Ferguson, J., Forrest, A., Williams, J., Hopkin, D., and Heard, G. (2010). 12 days under ice - an historic AUV deployment in the Canadian High Arctic. In *2010 IEEE/OES Autonomous Underwater Vehicles*, pages 1–11, Monterey, CA, USA.

Kaushal, H. and Kaddoum, G. (2016). Underwater optical wireless communication. *IEEE Access*, 4:1518–1547.

Kompella, V. R., Stollenga, M., Luciw, M., and Schmidhuber, J. (2017). Continual curiosity-driven skill acquisition from high-dimensional video inputs for humanoid robots. *Artificial Intelligence*, 247:313–335.

Korsah, G. A., Stentz, A., and Dias, M. B. (2013). A comprehensive taxonomy for multi-robot task allocation. *The International Journal of Robotics Research*, 32(12):1495–1512.

Kruijff, G. J. M., Pirri, F., Gianni, M., Papadakis, P., Pizzoli, M., Sinha, A., Tretyakov,

V., Linder, T., Pianese, E., Corrao, S., Priori, F., Febrini, S., and Angeletti, S. (2012). Rescue robots at earthquake-hit Mirandola, Italy: A field report. In *2012 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pages 1–8, College Station, TX, USA.

Kunz, C., Murphy, C., Singh, H., Pontbriand, C., Sohn, R. A., Singh, S., Sato, T., Roman, C., Nakamura, K.-i., Jakuba, M., Eustice, R., Camilli, R., and Bailey, J. (2009). Toward extraplanetary under-ice exploration: Robotic steps in the Arctic. *Journal of Field Robotics*, 26(4):411–429.

Lane, D. M., Maurelli, F., Kormushev, P., Carreras, M., Fox, M., and Kyriakopoulos, K. (2015). PANDORA - Persistent Autonomy Through Learning, Adaptation, Observation and Replanning. *IFAC-PapersOnLine*, 48(2):238–243.

LeHardy, P. K. and Moore, C. (2014). Deep ocean search for Malaysia Airlines flight 370. In *2014 Oceans - St. John's*, pages 1–4, St. John's, N.L., Canada.

Lesire, C., Infantes, G., Gateau, T., and Barbier, M. (2016). A distributed architecture for supervision of autonomous multi-robot missions. *Autonomous Robots*, 40(7):1343–1362.

Liu, M., Sivakumar, K., Omidshafiei, S., Amato, C., and How, J. P. (2017). Learning for multi-robot cooperation in partially observable stochastic environments with macro-actions. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1853–1860, Vancouver, Canada.

Ludvigsen, M., Albrektsen, S. M., Cisek, K., Johansen, T. A., Norgren, P., Skjetne, R., Zolich, A., Dias, P. S., Ferreira, S., Sousa, J. B. d., Fossum, T. O., Ø, S., Krogstad, T. R., Ø, M., Hovstein, V., and Vågsholm, E. (2016). Network of heterogeneous autonomous vehicles for marine research and management. In *OCEANS 2016 MTS/IEEE Monterey*, pages 1–7, Monterey, CA, USA.

Manley, J. E. (2007). The role of risk in AUV development and deployment. In *OCEANS 2007 - Europe*, pages 1–6, Aberdeen, Scotland, UK.

Matsuda, T., Maki, T., Sakamaki, T., and Ura, T. (2013). State estimation of multiple autonomous underwater vehicles for wide area survey of seafloor. In *OCEANS - Bergen, 2013 MTS/IEEE*, pages 1–9, Bergen, Norway.

Mausam and Kolobov, A. (2012). Planning with markov decision processes: An AI perspective. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(1):1–210.

McGann, C., Py, F., Rajan, K., Thomas, H., Henthorn, R., and McEwen, R. (2007). T-REX: A model-based architecture for AUV control. In *Proceedings of the Workshop on Planning and Plan Execution for Real-World Systems: Principles and Practices for Planning in Execution, ICAPS 2007*, Providence, RI, USA.

McPhail, S. D., Furlong, M. E., Pebody, M., Perrett, J. R., Stevenson, P., Webb, A., and White, D. (2009). Exploring beneath the PIG ice shelf with the Autosub3 AUV. In *OCEANS 2009-EUROPE*, pages 1–8, Bremen, Germany.

Meredith, M. P., Schofield, O., Newman, L., Urban, E., and Sparrow, M. (2013). The vision for a Southern Ocean Observing System. *Current Opinion in Environmental Sustainability*, 5(3):306–313.

Miloradović, B., Çürüklü, B., and Ekström, M. (2017). A genetic mission planner for solving temporal multi-agent problems with concurrent tasks. In *Advances in Swarm Intelligence, ICSI 2017*, Advances in Swarm Intelligence, pages 481–493, Fukuoka, Japan. Springer International Publishing.

Murphy, R. R., Dreger, K. L., Newsome, S., Rodocker, J., Slaughter, B., Smith, R., Steimle, E., Kimura, T., Makabe, K., Kon, K., Mizumoto, H., Hatayama, M., Matsuno, F., Tadokoro, S., and Kawase, O. (2012). Marine heterogeneous multirobot systems at the great Eastern Japan tsunami recovery. *Journal of Field Robotics*, 29(5):819–831.

Murphy, R. R., Steimle, E., Griffin, C., Cullins, C., Hall, M., and Pratt, K. (2008). Cooperative use of unmanned sea surface and micro aerial vehicles at Hurricane Wilma. *Journal of Field Robotics*, 25(3):164–180.

Muscettola, N. (1994). HSTS: Integrating planning and scheduling. *Intelligent Scheduling*.

Neerincx, M. (2003). Cognitive task load analysis: allocating tasks and designing support. In Hollnagel, E., editor, *Handbook of Cognitive Task Design*, pages 283–305. Lawrence Erlbaum Associates, Mahwah, NJ, USA.

Nelson, E. L., DeSoi, J. F., Mollenhauer, J. W., McClaran, S. R., Carroll, K. P., Pooch, U. W., and Williams, G. N. (1992). User interface design strategies for AUVC software development. In *Proceedings of the 1992 Symposium on Autonomous Underwater Vehicle Technology*, pages 152–157, Washington DC, USA.

Nunes, E., Manner, M., Mitiche, H., and Gini, M. (2017). A taxonomy for task allocation

problems with temporal and ordering constraints. *Robotics and Autonomous Systems*, 90:55–70.

Object Management Group (2017). OMG Unified Modeling Language (OMG UML). Standard formal/2017-12-05, Object Management Group.

Ocean Infinity (2018). MH370 operational search update #1 period 21-28 January 2018. Ocean Infinity, https://oceaninfinity.com/wp-content/uploads/MH370-Search-Weekly-Report-1.pdf. Accessed: February 2018.

Oliehoek, F. A. and Amato, C. (2016). *A concise introduction to decentralized POMDPs*, volume 1. Springer.

Orkin, J. (2004). Symbolic representation of game world state: Toward real-time planning in games. In *Proceedings of the AAAI Workshop on Challenges in Game Artificial Intelligence*, volume 5, pages 26–30, Menlo Park, CA, USA.

Pastore, T., Galdorisi, G., and Jones, A. (2017). Command and control (C2) to enable multi-domain teaming of unmanned vehicles (UxVs). In *OCEANS 2017*, Anchorage, AK, USA.

Pereira, A. A., Binney, J., Hollinger, G. A., and Sukhatme, G. S. (2013). Risk-aware path planning for autonomous underwater vehicles using predictive ocean models. *Journal of Field Robotics*, 30(5):741–762.

Pereira, F. L., Pinto, J., Sousa, J. B., Gomes, R. M. F., Goncalves, G. M., Dias, P. S., Pereira, F. L., Pinto, J., Sousa, J. B., Gomes, R. M. F., Goncalves, G. M., and Dias, P. S. (2006). Mission planning and specification in the Neptus framework. In *Proceedings 2006 IEEE International Conference on Robotics and Automation*, pages 3220–3225, Orlando, FL, USA.

Pettersson, L. (1997). Control system architectures for autonomous agents. Survey study, Department of Machine Design, Royal Institute of Technology, Stockholm, Sweden.

Picard, K., Brooke, B., and Coffin, M. (2017). Geological insights from Malaysia Airlines Flight MH370 search,. Eos, https://eos.org/project-updates/geological-insights-from-malaysia-airlines-flight-mh370-search. Accessed: June 2017.

Pinto, J., Faria, M., Fortuna, J., Martins, R., Sousa, J., Queiroz, N., Py, F., and Rajan, K. (2013). Chasing fish: Tracking and control in a autonomous multi-vehicle real-world experiment. In *2013 OCEANS - San Diego*, pages 1–6, San Diego, CA, USA.

Pinto, J., Ribeiro, M., Giodini, S., L'Hoir, P., Sartore, A., Giron-Sierra, J. M., Braga, J., and Sousa, J. (2017). Network enabled cooperation of autonomous vehicles: A communications perspective. In *OCEANS 2017 - Aberdeen*, pages 1–6, Aberdeen, Scotland, UK.

Py, F., Pinto, J., Silva, M. A., Johansen, T. A., Sousa, J., and Rajan, K. (2016). EUROPtus: A mixed-initiative controller for multi-vehicle oceanographic field experiments. In *International Symposium on Experimental Robotics*, pages 323–340, Tokyo, Japan.

Rajan, K., Py, F., and Barreiro, J. (2013). Towards deliberative control in marine robotics. In Seto, M. L., editor, *Marine Robot Autonomy*, pages 91–175. Springer New York, New York, NY.

Real-Arce, D. A., Morales, T., Bareera, C., Hernández, J., and Llinás, O. (2016). Smart and Networking UnderWAter Robots in Cooperation Meshes - The SWARMs ECSEL-H2020 Project. In *Seventh International Workshop on Marine Technology*, Barcelona, Spain. Martech.

Roehr, T. M., Cordes, F., and Kirchner, F. (2014). Reconfigurable Integrated Multirobot Exploration System (RIMRES): Heterogeneous modular reconfigurable robots for space exploration. *Journal of Field Robotics*, 31(1):3–34.

Russell, S. J. and Norvig, P. (2003). *Artificial Intelligence: A Modern Approach*. Pearson Education.

Saeedi, S., Trentini, M., Seto, M., and Li, H. (2016). Multiple-robot simultaneous localization and mapping: A review. *Journal of Field Robotics*, 33(1):3–46.

Sara Bernardini, D. E. S. (2007). Developing domain-independent search control for Europa2. In *Proceedings of the Workshop on Heuristics for Domain-independent Planning at ICAPS*, volume 7, Rhode Island, USA. Association for the Advancement of Artificial Intelligence (AAAI).

Sariel, S. (2007). *An Integrated Planning, Scheduling and Execution Framework for Multi-Robot Cooperation and Coordination*. PhD thesis, İstanbul Technical University.

Scerri, P., Farinelli, A., Okamoto, S., and Tambe, M. (2005). Allocating tasks in extreme teams. In *Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '05, pages 727–734, New York, NY, USA. ACM.

Schneider, T. and Schmidt, H. (2010). Unified command and control for heterogeneous marine sensing networks. *Journal of Field Robotics*, 27(6):876–889.

Schofield, O., Kohut, J., Aragon, D., Creed, L., Graver, J., Haldeman, C., Kerfoot, J., Roarty, H., Jones, C., Webb, D., and Glenn, S. (2007). Slocum gliders: Robust and ready. *Journal of Field Robotics*, 24(6):473–485.

Shukla, A. and Karki, H. (2015). Application of robotics in offshore oil and gas industry-a review part II. *Robotics and Autonomous Systems*, 75:508–524.

Soares, J. M., Aguiar, A. P., Pascoal, A. M., and Martinoli, A. (2013). Joint ASV/AUV range-based formation control: Theory and experimental results. In *2013 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5579–5585, Karlsruhe, Germany.

Somers, T. and Hollinger, G. A. (2016). Human–robot planning and learning for marine data collection. *Autonomous Robots*, 40(7):1123–1137.

Song, W., Dolan, J., Cline, D., and Xiong, G. (2015). Learning-based algal bloom event recognition for oceanographic decision support system using remote sensing data. *Remote Sensing*, 7:13564–13585.

Sotzing, C. C. (2009). *The Design and Implementation of a Multi-Agent Architecture to Increase Coordination Efficiency in Multi-AUV Operations*. PhD thesis, Heriot-Watt University.

Sotzing, C. C., Evans, J., and Lane, D. M. (2007). A multi-agent architecture to increase coordination efficiency in multi-AUV operations. In *OCEANS 2007 - Europe*, pages 1–6, Aberdeen, Scotland, UK.

Sotzing, C. C. and Lane, D. M. (2010). Improving the coordination efficiency of limited-communication multi–autonomus underwater vehicle operations using a multiagent architecture. *Journal of Field Robotics*, 27(4):412–429.

Spaan, M. T. J. (2012). Partially observable markov decision processes. In Wiering, M. and van Otterlo, M., editors, *Reinforcement Learning: State-of-the-Art*, pages 387–414. Springer Berlin Heidelberg, Berlin, Heidelberg.

Stokey, R., Austin, T., Von Alt, C., Purcell, M., Goldsborough, R., Forrester, N., and Allen, B. (1999). AUV bloopers or why Murphy must have been an optimist: A practical look at achieving mission level reliability in an autonomous underwater vehicle. In *International Symposium on Unmanned Untethered Submersible Technology*, pages 32–40.

Strutt, J. (2006). Report of the inquiry into the loss of Autosub2 under the Fimbulisen.

Technical report, N. Deposited at the request of Prof. Gwyn Griffiths on behalf of Autosub team.

Sutton, R. S., Precup, D., and Singh, S. (1999). Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112(1):181–211.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science*, 12(2):257–285.

Sydney, N. and Paley, D. A. (2014). Multivehicle coverage control for a nonstationary spatiotemporal field. *Automatica*, 50(5):1381–1390.

Toohey, L., Pizarro, O., and Williams, S. B. (2014). Multi-vehicle localisation with additive compressed factor graphs. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4584–4590.

Torreño, A., Onaindia, E., Komenda, A., and Štolba, M. (2017). Cooperative multi-agent planning: A survey. *ACM Computing Surveys*, 50(6):1–32.

Tsiogkas, N. and Lane, D. M. (2018). An evolutionary algorithm for online, resource constrained, multi-vehicle sensing mission planning. *arXiv e-prints*, page arXiv:1801.03552.

van Steen, M. and Tannenbaum, A. S. (2017). *Distributed Systems 3rd Edition*. Maarten van Steen, The Netherlands.

Verfuss, U., Aniceto, A., Biuw, M., Fielding, S., Gillespie, D., Harris, D., Jimenez, G., Johston, P., Plunkett, R., Siversten, A., Solbø, A., Storvold, R., and Wyatt, R. (2016). Literature review: Understanding the current state of autonomous technologies to improve/expand observation and detection of marine species. Technical report, SMRU Consulting.

Westwood, D. (2016). World auv market forecast 2016-2020. Douglas Westwood, http://www.douglas-westwood.com/report/oil-and-gas/world-auv-market-forecast-2016-2020/. Accessed: March 2018.

Williams, B. C., Nayak, P. P., and Nayak, U. (1996). A model-based approach to reactive self-configuring systems. In *Proceedings of AAAI-96*, pages 971–978, Portland, Oregon, USA.

Woods Hole Oceanographic Institute (2017). Underwater Wi-Fi: Testing the ABISS lander. WHOI, https://nautiluslive.org/video/2017/08/02/underwater-wifi-testing-abiss-lander. Accessed: November 2017.

Wynn, R. (2016). MASSMO showcase. In *NOC Marine Autonomy and Technology Showcase*, Southampton, England, UK. National Oceanography Centre.

Xargay, E., Dobrokhodov, V., Kaminer, I., Pascoal, A. M., Hovakimyan, N., and Cao, C. (2012). Time-critical cooperative control of multiple autonomous vehicles: Robust distributed strategies for path-following control and time-coordination over dynamic communications networks. *IEEE Control Systems*, 32(5):49–73.

Xiao, X., Dufek, J., Woodbury, T., and Murphy, R. (2017). UAV assisted USV visual navigation for marine mass casualty incident response. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 6105–6110, Vancouver, Canada.

Yuh, J., Marani, G., and Blidberg, D. R. (2011). Applications of marine robotic vehicles. *Intelligent Service Robotics*, 4(4):221.

Zeiler, M. D. and Fergus, R. (2014). Visualizing and understanding convolutional networks. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T., editors, *European Conference on Computer Vision*, pages 818–833, Zurich, Switzerland. Springer International Publishing.

Zlot, R. M. (2006). *An Auction-Based Approach to Complex Task Allocation for Multirobot Teams*. PhD thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA.

Chapter 3

Robust Mission Planning for Autonomous Marine Vehicle Fleets[1]

3.1 Introduction

This chapter proposes an integrated method of automated mission planning for a fleet of AMVs for long-term, large-scale missions such as inspection, maintenance, and monitoring of offshore structures. Mission planning for solo AMVs is non-trivial. Factors that need to be considered include the objectives that determine mission success, the requirements of the objectives, the suitability for the chosen vehicle to fulfil these requirements, and the associated risks and consequences that the vehicle and operator crew will be subject to during the mission. The complexities are compounded when multiple vehicles are deployed for different tasks, which places an increased cognitive load on the operators of the vehicles (Murphy et al., 2008). By implementing aspects of AI in the automation of mission planning such as task prioritisation, feasibility analysis, and path planning, the duties of the operator can be refocused on strategic objectives such as task generation and risk analysis.

Mission planning for AMVs can be structured into three procedural steps. The first step, referred to as *knowledge-based reasoning*, relies on the planning agent's (human operator or AI) knowledge base to identify tasks relevant to the mission objectives, requisite actions that will complete an identified task, and the sequencing of the identified tasks within a logical hierarchy of dependency succession (e.g. substructure must be cleaned of bio-fouling before inspection

[1]Sections 3.1-3.4.2 are a reproduction of the article submitted on the 14th May, 2019 to the journal Robotics and Autonomous Systems and is currently under review.

tasks can be performed). In AMV literature, HTNs (Sotzing and Lane, 2010, Lesire et al., 2016) and mixed-initiative planning (Ai-Chang et al., 2004, Py et al., 2016) have had the most success at providing easy methods for the operator to represent abstract tasks as sequences of primitive actions.

The second step, referred to as *task allocation*, is a common problem in multi-robot systems literature (Gerkey and Matarić, 2004). Task allocation is a multi-objective constraint satisfaction problem that considers the feasibility for a vehicle to complete the task (reliability), the vehicle's competence at completing the task to a required standard (quality), and the urgency of the task and the speed at which a vehicle can complete it (utility). To assess the suitability of a vehicle for a task, the task must first be refined down to a sufficient level of detail so that technical assessments can be made.

The third step is *risk projection* and requires the planning agent to consider uncertainties associated with allocated resources and mission structure within the context of failure analysis. The amount and magnitude of uncertainties are highly dependent upon the mission, the environment, and the vehicles involved and are difficult to generalise. If the mission proposal generated from steps one and two is considered acceptable in terms of risk and consequence, then the mission can be executed. Planners that can make projections on uncertainty provide the operator with decision support.

Deliberative planners exist in the marine robotics literature for both solo and fleet based operations. The EUROPtus mission planner (Py et al., 2016) was developed to assist operators with the scheduling and allocation of oceanographic sampling tasks to a variety of AMVs operating in meso-scale areas ($50km^2$). EUROPtus is a general purpose deliberative planner that uses temporal logic to allocate operator specified tasks to vehicles based on availability, producing functional plans that factor in task length uncertainty. Individual vehicles could then use their onboard T-REX planner (McGann et al., 2008) to repair plans *in situ*. This decentralised configuration is well suited for large sampling missions where vehicles do not have to directly cooperate to achieve mission objectives. The planner has not been trialled on missions that contain tasks with interdependencies. Additionally, the time domain is used to obtain mission plans but the planner itself does not consider the loadings the vehicle must overcome to complete the plan.

Following a HTN approach, (Sotzing et al., 2007) produced a task allocation method for mine countermeasure missions that distributes the mission plan onboard each vehicle. New tasks

that are generated by vehicles *in situ*, such as mine clearance tasks that result from positive mine detection tasks, are intermittently broadcasted using underwater acoustic communication, which will propagate to neighbouring vehicles. Because each vehicle is aware of the others, allocation is done locally by assessing the vehicle's proximity and ability to perform a new task against the others. This leads to inefficiencies in mission execution, where multiple vehicles may allocate themselves the same task based on out-of-date information. Giger (2010) developed mission planning tools that parallelise a task into smaller equivalent sub-tasks that can be allocated to individual AMVs. The genetic algorithm was used to allocate sub-areas to AMVs participating in a cooperative survey mission. The planning tool is limited by a uniform speed assumption and does not consider loadings on the vehicle that may reduce run times.

This chapter specifies a preprocessing procedure to formulate operator-specified tasks into the TOP (Chao et al., 1996). The TOP, which can be described as a combination of the vehicle routing problem (Dantzig and Ramser, 1959) and the binary knapsack problem (Kellerer et al., 2004), is formulated to be represented as a directed edge graph where the nodes represent locations of tasks that include a reward for being visited and the edges represent the cost of transitioning between two nodes. The objective of the TOP is to specify routes for multiple team members that maximise the combined reward and meet the individual cost constraints of the members.

The first preprocessing step clusters operator-specified task data and vehicle data into feasible operating zones for the vehicles based on their estimated Point of Safe Return, PSR, the furthest distance the vehicle can travel safely from a recharging station. For each cluster, collision-free transitions between nodes are determined using the Artificial Potential Field (APF) method established by Waydo and Murray (2003) and modified by Owen et al. (2012). A useful property of the vector field obtained by this particular APF is C^{∞} smoothness (Owen et al., 2012), meaning that the commanded velocity, acceleration, jerk or snap profiles can be directly determined according to the specifications of the vehicle controllers.

The next preprocessing step determines the energy consumption distribution of the vehicle by computing the required thruster output loads to follow the commanded velocity profile for the duration of the transition. Difficult to measure forces such as surface friction due to bio-fouling, thruster output variation due to non-linear voltage drop from battery discharge, wave induced pressure fluctuations, and localised currents can be accounted for by adding noise to and randomly sampling the parameters in the dynamics model. The mathematical model of the Reconfigurable Modular Robotic System for Aquatic Environment (REMORA) vehicle's

dynamics in calm water (Nielsen et al., 2018) was sampled and used in Monte Carlo simulation to produce the energy distribution. The path planner was tested for suitability with the control model of the REMORA AMV from Nielsen et al. (2018) and was shown to quickly stabilise and track the generated path.

With the data structured in terms of the TOP, a PSO variant formulated for discrete operations was chosen as the base solver. Discrete Strengthened Particle Swarm Optimisation (DStPSO) (Sevkli and Sevilgen, 2010) formulates PSO for the discrete domain and uses a reduced version of variable neighbourhood search as a local improvement to the global leader particle, strengthening the search. DStPSO was selected because it has few parameters, is simple to implement, and converges to near optimal global solutions in comparatively faster times than most other swarm optimisation methods (Ab Wahab et al., 2015). The original DStPSO method has been modified to include a linearly adaptive inertia weight based on the stall counter stopping criterion, and a swarm size decay algorithm (section 3.2.8) that prunes the swarm over each outermost loop of the worst performing particles.

The mission planner proposed in this chapter is the primary novel contribution to multi-AMV planning literature. The planner formulates the multi-AMV mission plan as the TOP rather than as a resource scheduling problem, with energy as the base planning resource instead of time. It is the use of energy as the planning resource that separates this mission planner from the current state-of-the-art planners (all temporal based). Energy resource optimisation is multi-objective in that it represents both the time taken and the loading on the vehicles, and requires dynamic models of the vehicles and the environment to provide a reliable plan. The planner is proposed as an alternative framework to existing temporal logic planners (Py et al., 2016) and HTN planners (Sotzing and Lane, 2010, Lesire et al., 2016) for marine vehicles.

The planner integrates several components that draw from separate fields of research: AI planning and operational research, marine vehicle systems, robot path planning, unsupervised learning, and Particle Swarm Optimisation (PSO). The integration of each of these components has resulted in minor contributions and improvements to existing literature in these fields as listed below:

1. A k-means clustering based algorithm that decomposes large-scale mission profiles into feasible operating zones, effectively eliminating large portions of the planner's search space.

2. A swarm size decay algorithm was added to an existing variant of PSO, DStPSO, to use

less computational resources while still achieving comparable quality in the generated plans.

3. The parameters within the marine vehicle dynamics model used for energy estimation were formulated as a stochastic process to accommodate uncertainties from real-world phenomena.

To evaluate the mission planner, a case-study inspection mission of the Anholt wind turbine array using a small fleet of REMORA AMVs was used as input data. Additionally, individual components of the planner were also given unit test instances: the hydrodynamic potential flow path planner was tested on the dynamic model REMORA and evaluated for reliability, and the swarm decay algorithm was evaluated in conjunction with DStPSO on 147 test datasets from Tsiligirides (1984). The results prove the planner is capable of producing solutions that maximise the number of visited turbines whilst adhering to the energy capacities of the vehicles according to the robust estimation of vehicle energy consumption.

3.2 Methodology

3.2.1 Problem Formulation

AMV fleet missions can be described abstractly as allocating and sequencing tasks to be completed at each target to the available vehicles. An applied mathematics problem that has a similar objective to the above is the TOP defined by Chao et al. (1996). In this problem, several agents must be allocated separate routes through a set of targets that represent tasks that yield a reward variable when completed. These routes must satisfy the energy constraints of the vehicles and maximise the collective reward of the team. An optimal solution for the TOP has the following characteristics:

1. Each vehicle has a unique set of tasks.
2. Each vehicle's route does not cross over itself.
3. Each vehicle's route has a predicted energy consumption that is close to the energy capacity of the vehicle.
4. Each vehicle starts and finishes at the nominated starting and finishing points.

3.2.2 Definition of the TOP Adapted for AMV Missions

The TOP stands as a solid representation for allocating sequences of points to team members. Variants of the TOP that consider time-windows (Labadie et al., 2012), stochastic weights (for the single vehicle OP) (Evers et al., 2014), time dependent weights (Li, 2011), and many others (see (Gunawan et al., 2016) for more variants) introduce aspects of real-world problems to the TOP. There are aspects of each of these variants that also suit AMV mission planning, but to begin with we present the following definitions that adapt the TOP to fit within the multi-robot systems and marine vehicle domains.

Definition 1. A task, T, is the tuple (g, s, I_t) where

- $g \in \mathbb{R}^3$ is the vector containing the location information of the task.
- $s \in \mathbb{R} \geq 0$ is the scalar reward yielded by completing the task.
- I_t is a tuple containing further information on the type of task, prerequisite tasks, and effects on other tasks.

Assumption 1. *For non-hierarchical missions (i.e. tasks are independent from each other), I_t simply points to the type of task, no prerequisite tasks or effects on other tasks need to be considered.*

Additionally, s is mapped to I_t by a time dependent reward function specified by the operator which must also depend upon the importance, urgency, and frequency of the task. We define such a reward function in section 3.2.5.

Definition 2. A vehicle, V, is the tuple (e_b, I_v) where

- $e_b \in \mathbb{R}$ is the energy storage capacity of the vehicle's batteries in Joules.
- I_v is a tuple containing further information on the vehicle identifier, type of vehicle, domain of operation, collision boundary, capabilities, and dynamic model.

Assumption 2. *For a homogeneous fleet (i.e. vehicles are of similar type and capability), I_v provides unique identifiers and the type, domain, capabilities and dynamic model are identical for all vehicles.*

Definition 3. An obstacle, O, is the tuple (X_o, r_o, I_o) where

- $X_o \in \mathbb{R}^3$ is the 3D position of the obstacle centroid.
- $r_o \in \mathbb{R}$ is the clearance radius the operator would like to maintain around the obstacle.
- I_o is a tuple containing further information on the obstacle, such as the classification (e.g. buoy, pile, rock, etc.) and the coordinate convention used by X_o.

Definition 4. The open mission, $\mathcal{M}_O$, is the sextuple $(\mathcal{T}, \mathcal{V}, \mathcal{O}, P, Q, E)$ where

- The operator defines N_T number of T which are collected in the N_T-tuple $\mathcal{T}$.
- The operator defines N_V number of V and collects them in the N_V-tuple $\mathcal{V}$.
- The operator defines N_O number of O and collects them in the N_O-tuple: $\mathcal{O}$.
- $P \in \mathbb{N}^{N_T}$ is the set of N_T sequential integers that references an element of $\mathcal{T}$: $P = \{1, \ldots, N_T\}$.
- $Q \in \mathbb{N}_V^N$ is the set of N_V sequential integers that references an element of $\mathcal{V}$: $Q = \{1, \ldots, N_V\}$.
- E is the zero-diagonal matrix of costs for transitioning between $\mathcal{T}_{P_i}$ and $\mathcal{T}_{P_j}$ and performing task $\mathcal{T}_{P_j}$: $E \in \mathbb{R}^{N_T \times N_T} \geq 0$.

$\mathcal{M}_O$ is the search domain of the planner. E is zero-diagonal because the transition $P_i = P_j$ is a forbidden transition. There are a total of $N_T^2 - N_T$ non-zero entries in E. The planner must provide a subset of $\mathcal{T}$ allocated to $\mathcal{V}$ as a proposal that can be evaluated for adherence to the energy constraints of the vehicles and the total reward yielded from completed tasks. The planner's proposal is specified as follows.

Definition 5. The closed mission, $\mathcal{M}_C$, is the quintuple $(\mathcal{T}, \mathcal{V}, R, S, F)$ where

- R is the N_V length set of tuples, where each tuple, R_Q, has an independent length $L_Q \geq 2$. R_Q is an ordered sequence subset of P corresponding to each vehicle's proposed route through $\mathcal{T}$.
- S is the set of rewards collected from completed tasks in $\mathcal{T}$: $S = \left\{ s \in \mathcal{T}_{R_Q} \right\}$.

- F is the N_V length set of tuples, each of length $L_Q - 1$, corresponding to the ordered sequence of elements of E accessed by the ordered sequential pairs in R_Q. $F_Q = \{E(R_{Q_i}, R_{Q_j}) \mid 1 \leq i \leq L_Q - 1,\ 2 \leq j \leq L_Q,\ (i,j) \in \mathbb{N}\}$.

The mission planner is then a solver that finds the most effective $\mathcal{M}_C$ according to the following fitness function and constraint:

$$\begin{aligned} &\underset{\mathcal{M}_C}{\text{maximise}} \quad \sum_{x_i \in S} x_i \\ &\text{subject to} \quad \sum_{y_i \in F_Q} y_i \leq e_b \in \mathcal{V}_Q \end{aligned} \tag{3.1}$$

To introduce real-world components to this framework, E is the sum of the energy consumed traversing from $g \in \mathcal{T}_{P_i}$ to $g \in \mathcal{T}_{P_j}$ along the collision free path $\mathcal{S}_{ij}$, labelled $E_{s,ij}$, the energy consumed completing the task at $\mathcal{T}_{P_j}$, labelled $E_{t,j}$ and the hotel load drain on the vehicle over the time taken to traverse the path and complete the task, labelled E_h. E is provided as the expectation of a stochastic process, which presents the opportunity of planning using stochastic weights. The models underlying the estimation of the energy variable are presented in section 3.2.3.

Because the vehicles must navigate around obstacles and take routes that are energy efficient, Euclidean distance calculations for $g \in \mathcal{T}_P$ may result in solutions that underestimate the actual distance travelled by the vehicle, which will subsequently underestimate the energy cost of traversing the path. Therefore, care must be taken with finding $\mathcal{S}_{ij}$. A simple method for projecting a smooth, collision free trajectory for an AMV is presented in section 3.2.7.

Similar to the TOP formulation, each vehicle must start and finish at two locations specified by the operator, which are inserted at the beginning and end of $\mathcal{T}$ as two special tasks, $\mathcal{T}_1$ and $\mathcal{T}_{N_T}$ respectively. For the *minimum operator effort* mission, we would like the vehicles to return to their deployment position, $\mathcal{T}_1 = \mathcal{T}_{N_T}$ is the special case called the *home point*. This is because in practice, AMVs are deployed from a central location such as a shore launch point, moored docking station or a vessel. The *home point* conveniently ensures the vehicles return to a position where they are able to recharge, offload collected data and diagnostic information, and be easily accessible for maintenance. In section 3.3.2, we describe a procedure to determine ideal location of $g \in \mathcal{T}_{\{1,N_T\}}$.

Solving the TOP is well studied and many solutions have been developed, most of which are available in (Gunawan et al., 2016). The DStPSO method (Sevkli and Sevilgen, 2012) was

selected as a meta-heuristic method for solving the TOP. Compared to other meta-heuristic solvers such as Ant Colony Optimisation, Genetic Algorithm, and Tabu search, DStPSO reaches near-optimum solutions faster with fewer parameters (Ab Wahab et al., 2015).

To summarise, the following components from $\mathcal{M}_O$ are required in order to be solved in a similar fashion to the TOP:

1. The number of tasks, N_T
2. The number of vehicles, N_V
3. Each vehicle's battery energy storage constraint, e_b.
4. The value of each task's reward, s.
5. The cost of moving to and performing a task, E_{ij}.

The TOP requires the mission data to be processed into the above form to find an optimum $\mathcal{M}_C$, which is distributed to the vehicles upon deployment. The proposed planner follows the process in figure 3.1. The following sub-sections detail the steps taken to obtain each of the components of the process.

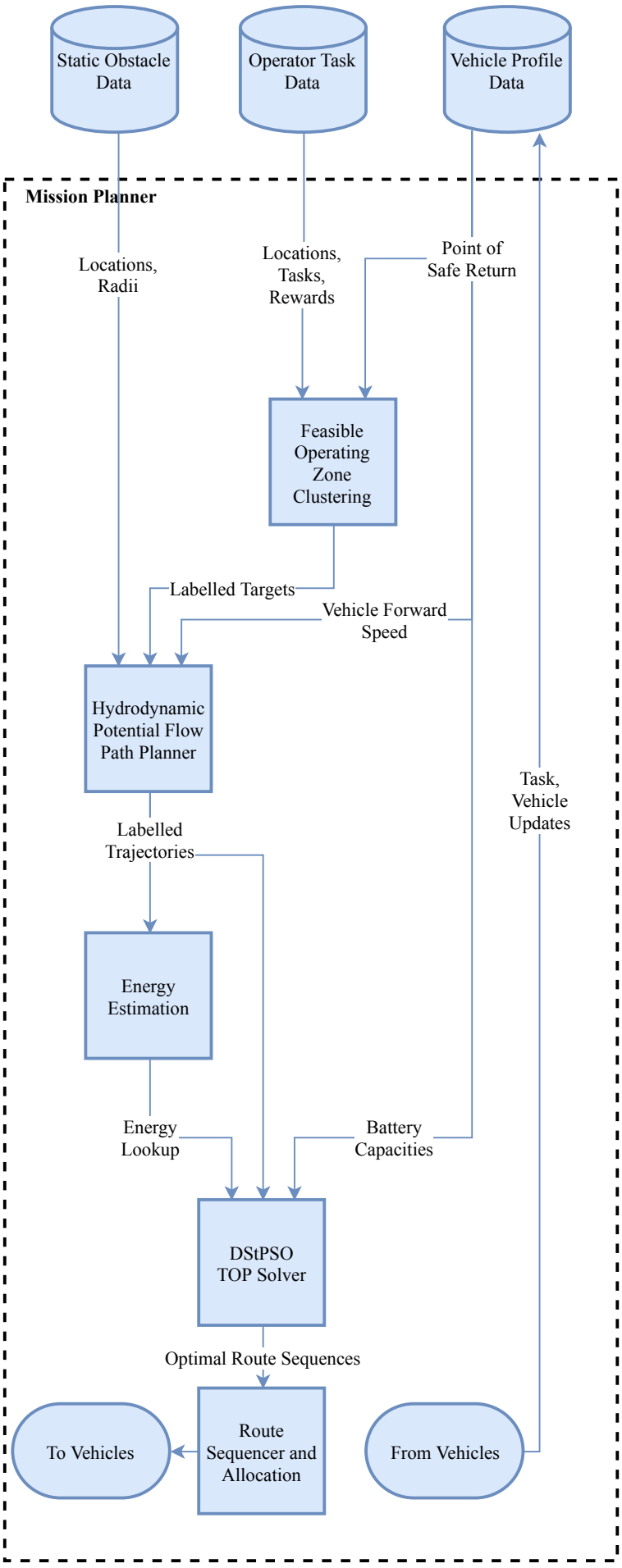


Figure 3.1: Process flow of the proposed mission planner.

3.2.3 Energy Consumption Prediction

Let $\mathcal{S}_{ij}$ be the collision free path connecting the task location $g \in \mathcal{T}_{P_i}$ to $g \in \mathcal{T}_{P_j}$. The energy consumed by the vehicle $V \in \mathcal{V}$ to traverse the route and perform the tasks is given by

$$E_{ij} = E_{s,ij} + E_{t,j} + E_h, \quad i \neq j \tag{3.2}$$

where $E_{s,ij}$ is the energy spent to traverse the path, $E_{t,j}$ is the energy depleted to perform the task T at $g \in \mathcal{T}_{P_j}$ and E_h is the energy consumed by the hotel load.

The energy required to compensate drag, centripetal and buoyancy forces/moments while traversing the path $\mathcal{S}_{ij}$ is

$$E_{s,ij} = \int_{\mathcal{S}_{ij}} \boldsymbol{\tau}(\mathbf{s})\, \mathrm{d}\mathbf{s}, \tag{3.3}$$

where $\boldsymbol{\tau} \in \mathbb{R}^6$ is the generalised vector of control forces and moments acting on the vehicle and $\mathbf{s} \in \mathbb{R}^6$ is the path variable. For a vehicle outfitted with N_{th} thrusters the generalised vector $\boldsymbol{\tau}$ is provided as the linear combination of the thrust command vector $\mathbf{t} \in \mathbb{R}^{N_{th}}$ through the thruster configuration matrix $\mathbf{T}_c \in \mathbb{R}^6 \times \mathbb{R}^{N_{th}}$

$$\boldsymbol{\tau} = \mathbf{T}_c \mathbf{t}. \tag{3.4}$$

Let $\boldsymbol{\eta} = [N, E, D, \phi, \theta, \psi]^{\mathrm{T}} \in \mathbb{R}^6$ be the generalised pose vector of the vehicle in the inertial North-East-Down (NED) frame and $\boldsymbol{\nu} = [u, v, w, p, q, r]^{\mathrm{T}} \in \mathbb{R}^6$ be the generalised linear and angular velocity vector in a body-fixed frame. For a vehicle manoeuvring at constant speed ($\boldsymbol{\nu} = \bar{\boldsymbol{\nu}}$) the generalised control forces and moments balance the hydrostatic and hydrodynamic contributions, i.e.

$$\bar{\boldsymbol{\tau}} = \mathbf{D}(\bar{\boldsymbol{\nu}})\bar{\boldsymbol{\nu}} + \mathbf{C}(\bar{\boldsymbol{\nu}})\bar{\boldsymbol{\nu}} + \mathbf{g}(\bar{\boldsymbol{\eta}}) \tag{3.5}$$

where $\mathbf{D}(\boldsymbol{\nu}) \in \mathbb{R}^6 \times \mathbb{R}^6$ is the linear plus quadratic drag; $\mathbf{C}(\boldsymbol{\nu}) \in \mathbb{R}^6 \times \mathbb{R}^6$ accounts for Coriolis and centripetal forces and moments; $\mathbf{g}(\boldsymbol{\eta}) \in \mathbb{R}^6$ is the vector of restoring moments and forces. $\bar{\boldsymbol{\tau}}$ represents an underestimate of the total generalised force spent to traverse the path since it does not account for acceleration and deceleration phases.

The electrical power spent by each thruster to deliver the thrust t_i is generally approximated with a quadratic function of the commanded thrust t_i (Furno et al., 2017), i.e.

$$\Pi_k = \eta_k |t_k| t_k, \quad k = 1, \ldots, N_{th} \tag{3.6}$$

where η_k is a thrust efficiency coefficient. Therefore the energy required to traverse the path $\mathcal{S}_{ij}$ can be computed as

$$E_{s,ij} = \int_{\mathcal{S}_{ij}} \bar{\boldsymbol{\tau}}(\mathbf{s})\, \mathrm{d}\mathbf{s} = \int_{\mathcal{S}_{ij}} \mathbf{T}_c \bar{\mathbf{t}}(\mathbf{s})\, \mathrm{d}\mathbf{s} = \int_{t_i}^{t_j} \mathcal{P}(t)\, \mathrm{d}t, \tag{3.7}$$

where $\mathcal{P} = \sum_k \Pi_k$ is the total electric power consumed to traverse the path $\mathcal{S}_{ij}$, t_i is the time instant the vehicle leaves $g \in \mathcal{T}_{P_i}$ and t_j is the time instant the vehicle arrives to $g \in \mathcal{T}_{P_j}$.

Equation 3.7 shows that the energy $E_{s,ij}$ is function of the hydrostatic and hydrodynamic characteristics of the specific vehicle V through Equation 3.5. The coefficients of $\mathbf{D}(\boldsymbol{\nu})$, $\mathbf{C}(\boldsymbol{\nu})$ and $\mathbf{g}(\boldsymbol{\eta})$ are usually estimated from model tests and hence affected by uncertainty (Nielsen et al., 2018). Furthermore, as the vehicle performs missions natural wear and tear will affect its hydrodynamic characteristics resulting in changes of the coefficients. This implies that a pure deterministic description of $\boldsymbol{\tau}$ may cause severe underestimates of the energy consumption associated with traversing the route R_Q. To account for model uncertainties and wear and tear the steady state generalised vector $\bar{\boldsymbol{\tau}}$ is modelled as a Gaussian random vector with mean $\boldsymbol{\mu}_\tau$ and covariance matrix $\boldsymbol{\Sigma}_\tau$, i.e. $\bar{\boldsymbol{\tau}} \sim \mathcal{N}(\boldsymbol{\mu}_\tau, \boldsymbol{\Sigma}_\tau)$. Therefore both the power spent and the energy needed to traverse the path become random variables and their estimates are computed through the expected value operator $E[\cdot]$ as

$$\hat{E}_{s,ij} = E[E_{s,ij}] = E\left[\int_{\mathcal{S}_{ij}} \bar{\boldsymbol{\tau}}(\mathbf{s})\, \mathrm{d}\mathbf{s}\right] = \int_{\mathcal{S}_i ij} \mathbf{T}_c E[\bar{\mathbf{t}}(\mathbf{s})]\ \mathrm{d}\mathbf{s} = \int_{t_s}^{t_f} E[\mathcal{P}(t)]\ \mathrm{d}t. \tag{3.8}$$

Remark 1. Since the matrices $\mathbf{D}(\boldsymbol{\nu})$, $\mathbf{C}(\boldsymbol{\nu})$ and the vector $\mathbf{g}(\boldsymbol{\eta})$ are linear in the parameters and each parameter is estimated as being normally distributed, then the generalised vector of forces and moments is normally distributed. However as the vehicle ages through operations the wear and tear may determine changes in the parameters such that distributions other than normal will be better suited. This implies that model parameters should be periodically re-estimated in order to reduce errors in the energy consumption estimation.

In addition to expending energy while transitioning between task locations, the vehicle also expends energy undertaking a particular task at location $g \in \mathcal{T}_{P_j}$. Tasks vary in energy intensity. For example, a vehicle tasked with cleaning substructure from bio-fouling will experience higher loads than a vehicle tasked with visual inspection of the same substructure, meaning a higher energy consumption for the former scenario. Therefore an estimate of the task energy $E_{t,j}$ can be computed after the specific definition of the tuple $I_t \in \mathcal{T}_{P_j}$. For example an inspection task

as a seabed survey or a scrutiny of a monopile will be defined as a trajectory and a sequence of actions to be performed while passing at given way points. The energy cost associated with the trajectory tracking will then be estimated by means of equation 3.8; while the energy spent in carrying out the sequence of actions will be evaluated based on the sensors and actuators to be used and the usage duration.

The energy depleted by the hotel load $\mathcal{H}$ is usually accounted for by considering the nominal power consumption of the guidance, navigation, control, communication, environmental sensing and acting systems that are switched on during the mission. An energy baseline for the hotel load can be estimated by considering those systems that must always be available, i.e. the guidance, navigation and control computer with associated sensors and the communication system. Instead of looking into component data sheets for the nominal power consumption declared by the manufacturers, the hotel load $\mathcal{H}$ can be modelled as a random variable by looking into logged data while the vehicle is idle. For the considered REMORA vehicle study case, recorded data of power consumption shows that the baseline hotel load can be modelled as a normally distributed random variable, i.e. $\mathcal{H} \sim \mathcal{N}(\mu_{\mathcal{H}}, \sigma^2_{\mathcal{H}})$. Hence the energy cost of the hotel load is given by

$$\hat{E}_h = E[E_h] = E\left[\int_{t_{s,k}}^{t_{t,k}} \mathcal{H}\,\mathrm{d}t\right] = \int_{t_{s,k}}^{t_{f,k}} E[\mathcal{H}_k]\,\mathrm{d}t \tag{3.9}$$

where $t_{s,k}$ and $t_{f,k}$ are the start time of the transition $g \in \mathcal{T}_{P_i} \rightarrow g \in \mathcal{T}_{P_j}$ and the finish time of the task $\mathcal{T}_{P_j}$, respectively.

Remark 2. As the vehicle executes the mission, different sensors and actuators are powered up in order to fulfil the assigned tasks. This will generate power loads that may change the statistical description of the hotel load towards non-symmetric distributions with heavy tails (e.g. Rayleigh distribution).

3.2.4 Vehicle Range

The range of a vehicle depends on its total energy storage, hotel load, power distribution efficiency, mechanical efficiency, propulsive efficiency, hydrodynamic drag properties, and environmental loadings. Estimating all of these properties, which in reality vary with time, is non-trivial. However, an ideal range for a specified forward speed can be obtained based on approximated constants as described by Furlong et al. (2012). For most vehicles, the range is obtained through endurance testing under certain speeds and weather conditions (Hobson et al.,

2012). The range of the vehicle can be inferred for new conditions based on this knowledge.

In section 3.2.3 we proposed a stochastic approach as an alternative to the above methods to obtain an energy consumption distribution for the vehicle over a velocity profile. We can use the steady state power consumption evaluated for the vehicle travelling at a constant forward speed to estimate the forward distance travelled before the vehicle's energy storage, e_b, is depleted. The battery's capacity (C_b, measured in Ah) and its nominal operating voltage, V_b, are related to its energy storage by:

$$e_b = C_b \times 3600(s/h) \times V_b \tag{3.10}$$

The point-of-safe-return, PSR, is useful for robust mission planning because it puts an upper bound on the distance the vehicle is allowed to be from its home point. The PSR for the vehicle operating at a constant forward speed of $U = \bar{U}$ m/s is:

$$PSR = \frac{\bar{U} \times E_b}{2(\mathcal{P} + \mathcal{H})} \tag{3.11}$$

The difficulty in this procedure is that the energy consumption is non-trivial to predict in the marine environment. Dynamic loads from waves, wind, tide, current, thermal currents, and hydrodynamic forces all influence the effort generated by the vehicle's thrusters. Hydrodynamic effects such as turbulence, influenced by bio-fouling and surface degradation, and the design of the vehicles thrusters also affect the efficiency of the vehicle, dependent upon the vehicle's speed and thruster RPM. Small-scale hydrodynamic effects can be captured through the use of parameter variation in the dynamic model. However, large-scale effects such as current, waves, and wind must be added as separate estimator components to the base dynamic model.

3.2.5 Modelling Rewards

For the TOP to be solved, rewards for completing a task must be assigned to each target. In the case of recurring tasks such as maintenance and cleaning, reward is primarily a function of time since the task was last completed. Machine learning methods such as linear regression could be used to estimate the reward of a task from a sensor based data set (such as measuring the vibration of a structure). As a simple, parameterised alternative, the sigmoid function, a popular continuous activation function in machine learning, is selected as the candidate function

for representing the reward of a task:

$$s(t) = \frac{1}{1 + e^{-t}} \tag{3.12}$$

The range of the sigmoid function is $[0, 1]$, hence it is useful as a binary activation switch. However, it is desired that tasks can be parameterised in terms of importance, importance growth-rate, and frequency. The generalised logistic function, attributed to Richards (1959), could allow the operator increased flexibility with how reward $s \in T$ grows or decays with time. Given the generalised logistic function:

$$s(t) = A + \frac{K - A}{(C + De^{Bt})^{1/\delta}} \mid (A, B, C, D, K, \delta) \in \mathbb{R} \tag{3.13}$$

The operator can control the start and end values with A and K, and rate of growth/decay B of $s \in T$ (see figure 3.2). As time progresses and T has not been completed, s can grow or decay. The independent variable, t, can be set to 0 upon completion of a task to restart the reward function.

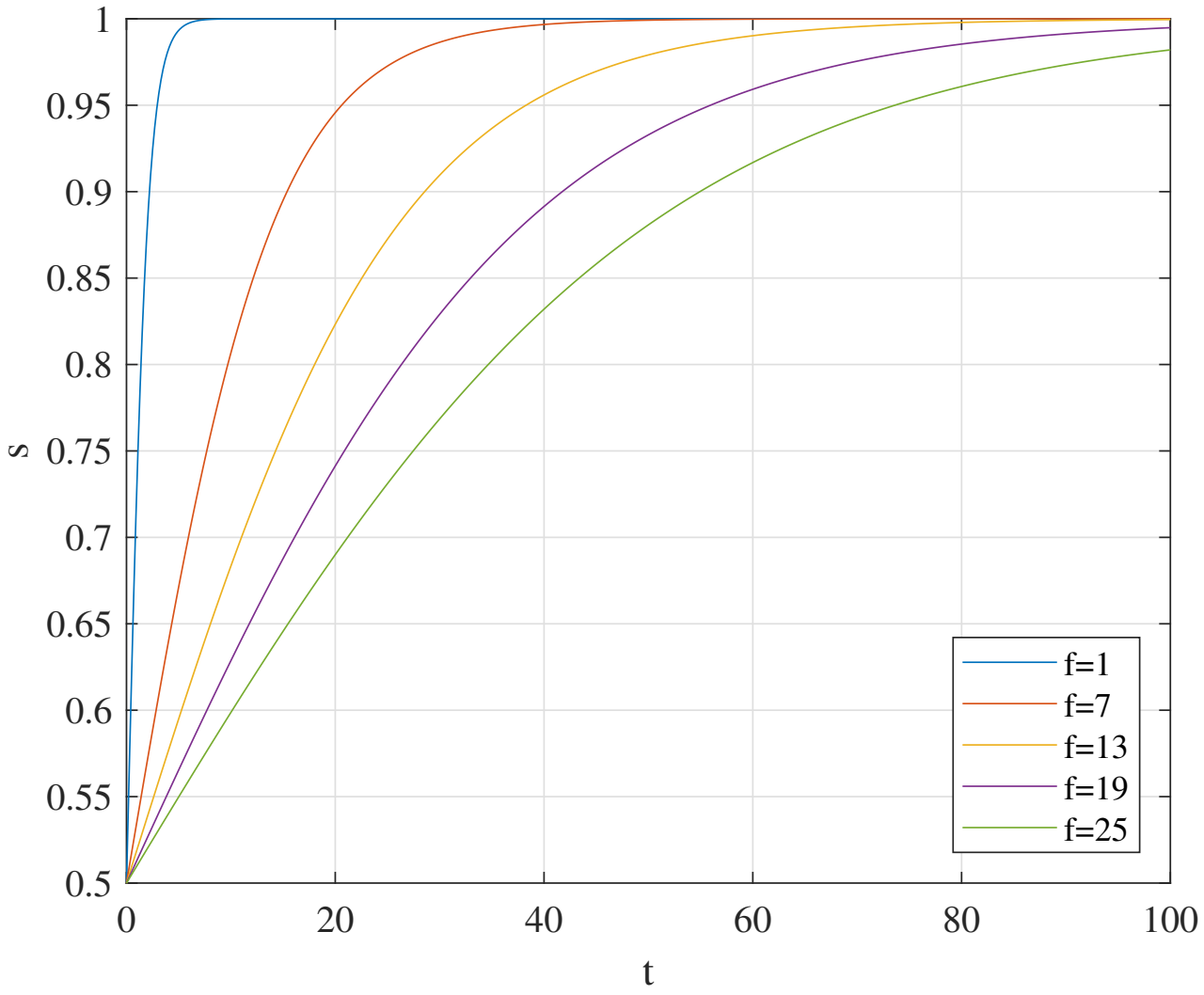


Figure 3.2: Generalised Sigmoid Reward Over Arbitrary Timescale, $A = 0$, $B = \frac{1}{f}$,$C = 1$, $D = 1$, $K = 1$ and $\delta = 1$

3.2.6 Target Clustering

The selection of points available for the vehicle team to visit must all be within the PSR of the vehicle with the largest range. For target sets that are distributed over large areas, such as offshore wind farm installations or macroscale marine sampling, some of the targets will always be outside of any of the vehicles' reach for any time instance. Problems that contain many targets will have larger search spaces and will take longer to solve. Removing the infeasible targets will simplify the search domain. We propose grouping target sets into clusters that are sized appropriately so that they are within serviceable range of the vehicles from the centroid of the cluster. k-means clustering is a suitable method for obtaining appropriately sized clusters of targets. Algorithm 1 details a simple procedure that achieves feasible operating zones via

target clustering.

Algorithm 1: Feasible operating zone clustering

input : Target coordinates $L \in \mathbb{R}^{N\times 3}$; Point of Safe Return PSR

output: C sets of indexes, X_best centroids of each cluster

1 $flag = 0$;
2 $N_C = 0$;
3 **while** $\neg flag$ **do**
4 $N_C \leftarrow N_C + 1$;
5 $id \leftarrow$`zeros(`$N_C, 1$`)`;
6 $sumD_{best} \leftarrow 0$;
7 **for** $i \leftarrow 1$ **to** $reps$ **do**
8 $[IDX, X, sumD, sqdD] \leftarrow$`kmeans(`$L, N_C$`)`;
9 **if** $1/sumD > sumD_{best}$ **then**
10 $sumD_{best} \leftarrow 1/sumD$;
11 $IDX_{best} \leftarrow IDX$;
12 $X_{best} \leftarrow X$;
13 $sqdD_{best} \leftarrow sqdD$;
14 **for** $i \leftarrow 1$ **to** N_C **do**
15 $id(i) \leftarrow PSR^2 >$ `max(`$sqdD_{best}(IDX_{best} == i)$`)`;
16 $flag \leftarrow$`all(`id`)`;
17 **for** $i \leftarrow 1$ **to** N_C **do**
18 C$(i) \leftarrow IDX_{best}(IDX_{best} == i)$;
19 **return** C,X_{best};

The set of locations for all non-special tasks, $\{g \in \mathcal{T}_i \mid 2 \leq i \in \mathbb{N} \leq N_T - 1\}$, and the largest calculated PSR of the vehicles are used as inputs to the algorithm. Lines 7-13 replicate the k-means clustering function on the location data *reps* times, the solution with the best fit (i.e. the lowest $sumD$) is chosen. Lines 14-16 checks that the point furthest from center in each cluster is less than the specified PSR. If this constraint is not met, then the number of required clusters (N_C) is increased and the process begins again. The returned variable, C, is the tuple of length N_C where each element corresponds to a unique subset of P. C is used to subdivide $\mathcal{M}_O$ into N_C sub-missions, labelled as $\mathcal{M}_O^{(k)}$ where $1 \leq k \in \mathbb{N} \leq N_C$. The special *home point* tasks $\mathcal{T}_{\{1,N_T\}} \in \mathcal{M}_O^{(k)}$ have their location set to $X_{best}^{(k)}$, the centroid of the corresponding k-th

cluster in $\mathbb{R}^3$.

3.2.7 Obtaining Paths

Assuming straight-path distances between targets will lead to underestimation in energy costs when planning routes for vehicles in environments containing obstacles. The vehicle will occasionally take non-straight paths, either as a result of navigating around obstacles or because the dynamics of the vehicle prevent it from instantaneously adjusting to the reference trajectory, and a planner that does not account for this may produce optimistic plans that are unattainable by the vehicle. The planner must have a realistic estimate of the distance of a collision free and dynamically viable path, which is a well studied problem in robot path planning literature (Mac et al., 2016).

Generating a valid path for a vehicle to transition from one point to another requires consideration of the obstacles between the vehicle's starting and finish points for a transition. For a basic static obstacle avoidance method, the following components are required:

1. Vehicle's starting location and destination, $\{g \in \mathcal{T}_{\{i,j\}}\}$.
2. Vehicle's collision radius, defined in $I_v \in V$.
3. Static obstacle locations and sizes, $\mathcal{O}$.

There are many successful methods available in path planning literature: Probabilistic Road Maps (Kavraki et al., 1996), Rapidly exploring Random Tree (Kuffner and LaValle, 2000), A* (Brooks and Lozano-Pérez, 1985), any-angle (Θ^*) (Choi et al., 2010), and APF (Khatib, 1985). APF methods that use hydrodynamic potential flow theory ((Waydo and Murray, 2003, Pedersen and Fossen, 2012)) can produce smooth, spline-like trajectories efficiently because the search domain is defined in part by analytic equations. Pedersen and Fossen (2012) developed a particle pursuit guidance controller for marine vehicles that used the stream function of a hydrodynamic APF to guide a vessel around circular obstacles, but could not guarantee that the particle would not cross an obstacle boundary. Circular obstacles were modelled as a potential field using the circle theorem (Milne-Thomson, 2013) that guarantees zero boundary crossflow, which was used for APF path planning for UAVs by Waydo and Murray (2003). From the definition of O in section 3.2.2, the circle theorem APF method suitably fits as a base path planning model within the AMV mission planner framework. We have adapted this method to generate collision-free routes for marine vehicles.

If the position and velocity of the vehicle is represented in the complex domain $\mathbb{C}$ respectively by

$$z = x + iy \tag{3.14}$$

$$\frac{dz}{dt} = u + iv \tag{3.15}$$

where $\{x, y, u, v\} \in \mathbb{R}^4$ are referenced to the planar world frame. The Partial Complex Velocity (PCV) flow field, as derived from the circle theorem used in (Waydo and Murray, 2003, Owen et al., 2012), is:

$$\frac{dz}{dt} = \frac{Q_s}{2\pi(z-c)} + \frac{Q_s}{2\pi}\frac{r^2}{(b-z)(r^2+(b-z)(\bar{c}-\bar{b}))}, \tag{3.16}$$

where Q_s is the strength of the source ($Q_s > 0$) or the sink ($Q_s < 0$), $c \in \mathbb{C}$ is the location of the source/sink (the starting or finishing point), the radius of the obstacle $r \in O$ and $b \in \mathbb{C}$ is the complex variable of $X_o \in O$ in the X-Y plane. The full Complex Velocity (CV) field for an obstacle is the sum of the sink and source PCV fields. For multiple obstacles, simply summing the CV fields will not produce a valid field that represents all of the obstacles. As discussed in (Pedersen and Fossen, 2012), the cross flow at the boundary of each obstacle is influenced by the CV flows of all other obstacles. In Waydo and Murray (2003), Owen et al. (2012), these influences are eliminated at each obstacle boundary by introducing a weighting term for each obstacle's CV:

$$\alpha_i = \prod_{j \neq i} \frac{d_j^4}{d_i^4 + d_j^4}, \tag{3.17}$$

where d_i and d_j are the Euclidean distances between the vehicle's current position z and the i-th and j-th obstacle centroids. The complete CV flow for N_O obstacles is then:

$$\text{CV} = u + iv = \sum_{i=1}^{N_O} \alpha_i (\text{PCV}_i^{source} + \text{PCV}_i^{sink}) \tag{3.18}$$

In effect α_i interpolates the CVs of each obstacle with a weighting from 0 to 1, ensuring that the obstacle closest to the vehicle will have an increasingly dominant flow compared to the other obstacles.

Equation 3.18 represents a first order differential equation that can be integrated to obtain the path of the vehicle from a given initial condition. The Runge-Kutta Dormand-Prince (RKDP) method was selected to evaluate equation 3.18 given a set of obstacles, obstacle radii, and the vehicle's initial and final positions. Compared to the Euler method used by Owen et al.

(2012), RKDP can solve long trajectories (>1000s) extremely quickly by adapting the step size to minimise calculations whilst retaining an acceptable error tolerance from the real solution.

The method in Owen et al. (2012) was developed for non-holonomic vehicles by offsetting the location of the source behind the vehicle position. For holonomic vehicles (i.e. vehicles that can turn on the spot such as ROVs and hovering AUVs), several orientations can be searched through from a given starting position using a fitness function to evaluate each solution for shortest travel time, vehicle dynamics, safety, and efficiency. We have used a simple fitness function to determine the shortest path:

$$Z = \frac{1}{(t_f - t_0)} \tag{3.19}$$

The highest scoring solution will have the shortest path. This ensures that the least energy consuming path is taken given the assumption that the environment is ideal and that the vehicle can accurately follow the underlying velocity profile. In practice the shortest path is typically the starting orientation $\psi_0 = \text{atan2}(y_{sink} - y_0, x_{sink} - x_0)$, but if there are many obstacles along this path, other orientations may yield shorter routes. The REMORA's holonomic underwater vehicle model was tested in simulation for following a path generated by integration of equation 3.18 and is presented in section 3.2.7.

3.2.8 Proposal Generation

Our implementation of DStPSO (pictured in figure 3.3) follows the same principles of PSO but has been adapted to work in the discrete domain, strengthened with a local search heuristic on the pioneering particles, and a swarm decay heuristic to save on computational resources. As described in section 3.2.2, the search space for the DStPSO algorithm is restricted to $\mathcal{M}_O$. We define a particle by its position $R \in \mathcal{M}_C$ and velocity W. W is the set of points in P that are not in any element of R: $W = P \setminus R$.

From equation 3.1, the position of a particle is subject to the energy constraints of the vehicles. By obtaining F, the feasibility of a route can be determined by checking:

$$\sum_{y_i \in F_i} y_i \leq e_b \in \mathcal{V}_i \forall i \in Q \tag{3.20}$$

At its core, DStPSO updates its position by inserting random elements from W into elements of R using various insertion method heuristics, constrained by the above energy relation.

A particle is initialised by setting each element of R to $\{1, N_T\}$, corresponding to the special *home point* tasks $\mathcal{T}_1$ and $\mathcal{T}_{N_T}$. The velocity is then $W = P \setminus R = P \setminus \{1, N_T\}$. Each element of R is then sequentially modified by iteratively selecting a random element from W, inserting it using the *cheapest insertion heuristic* (Mester et al., 2007), and keeping the solution if the updated F still meets the energy constraint. W has the selected element removed and the process repeats until all elements of W have been tried.

The swarm, $\mathcal{Q}$ is the set of N_Q initialised particles. Each particle in $\mathcal{Q}$ is evaluated for fitness by finding the total collected reward for its current position:

$$\sum_{x_i \in S} x_i \forall i \in R_Q \tag{3.21}$$

The N_Q long set of particle positions, *pbest*, is initialised by setting each element of *pbest* equal to the position of the corresponding particle in $\mathcal{Q}$. *pbest* keeps a running record of the highest scoring position that each particle has visited. $pbest_i$ is only updated when Q_i moves to a position with a fitness higher than the corresponding score of *pbest*. Finally, the particle that has the highest fitness out of *pbest* is assigned to *gbest*. *gbest* is only updated if the fittest particle in the updated *pbest* is higher than the fitness of the current *gbest*.

When *gbest* is updated, a local search is triggered on *gbest* using a simplified version of Variable Neighbourhood Search (VNS) (Hansen et al., 2010) called Reduced Variable Neighbourhood Search (RVNS) (Sevkli and Sevilgen, 2010). RVNS implements three heuristic search methods (or neighbourhoods) on *gbest*: 1. *insert for increasing profit*, 2. *insert for decreasing cost*, and 3. *path inversion* (also known as 2-opt (Croes, 1958)). Each neighbourhood is evaluated for feasibility and improvement, and if the new position meets both criteria then the neighbourhood is set back to neighbourhood 1. If neighbourhood 3 fails to improve the solution several consecutive times, RVNS returns the updated *gbest* and the particle that had pioneered *gbest* is reinitialised to encourage exploration. For further details on RVNS, see Sevkli and Sevilgen (2010).

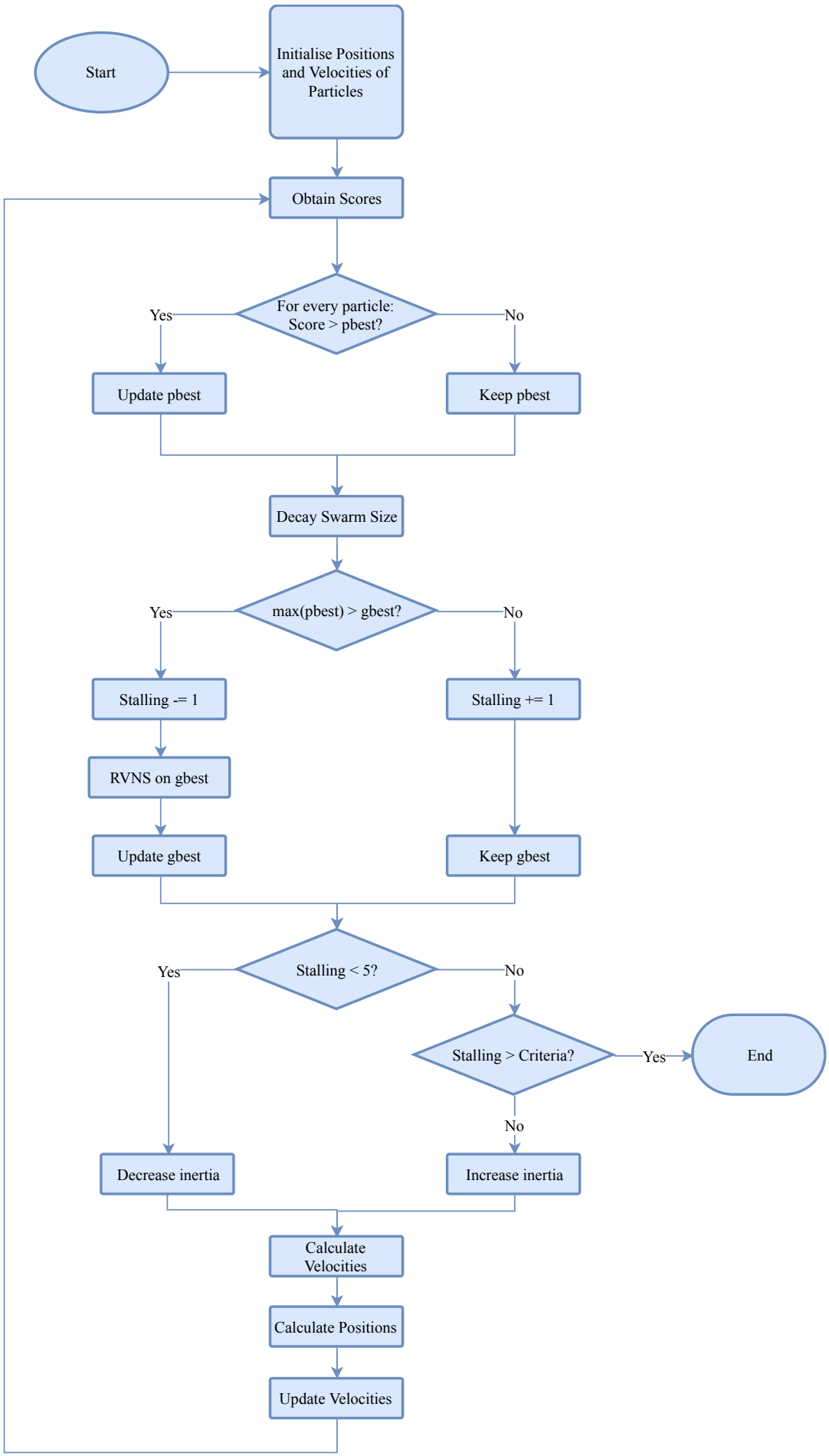


Figure 3.3: DStPSO procedure with swarm decay.

After the *gbest* local search, or if no *gbest* update occurs, the velocity of the i-th particle in $\mathcal{Q}$ is updated in a manner similar to the original PSO,

$$W_i = [w \otimes W_i] \oplus [(c_1 \otimes (pbest_i \ominus R_i)) \oplus (c_2 \otimes (gbest \ominus R_i))], \tag{3.22}$$

where w represents the typical inertia term used in PSO, and c_1 and c_2 are weighting terms that balance exploration between the particle's best experience and the swarm's best experience. The position and the velocity are subsequently updated by

$$R_i = R_i \circ W_i \tag{3.23}$$

$$W_i = P \ominus R_i, \tag{3.24}$$

where each of the special operators, $\otimes$, $\oplus$, $\ominus$, and $\circ$ are defined as follows:

- $\otimes$ Each element of the right hand side (RHS) of the operator is given a random number from 0 to 1. The left hand side (LHS) is a scalar number between 0 and 1. The output is the subset of the RHS that is less than the LHS.
- $\oplus$ Combines two velocity sets. If the LHS contains the *pbest* term and the RHS contains the *gbest* term, then the output is the reordered set $\{RHS \cap LHS, RHS \setminus LHS, LHS \setminus RHS\}$. Otherwise the output is the reordered set $\{RHS, LHS\}$.
- $\ominus$ Is the set difference $LHS \setminus RHS$.
- $\circ$ Apply *insert for increasing profit* from RVNS neighbourhood 1 on the RHS velocity set to the position set on the LHS.

The DStPSO terminates when no successor to the current *gbest* is found for a consecutive number of iterations. The proposed set of routes for each vehicle, $R \in \mathcal{M}_C$ is set to *gbest*.

Improvement to DStPSO with Swarm Size Decay

As the swarm size increases, so too does the exploratory power of DStPSO and the computational resources required for particle position updates. A balance between these two outcomes can be exploited by starting out with a large N_Q compared to what is used in practice (usually between 10 to 40 particles for solving the TOP), and then reducing the size of $\mathcal{Q}$ on each iteration by keeping the best performing particles until a minimum size is reached. With this modification, DStPSO begins with a wide exploration of the solution space, providing a better

chance of pioneering a near optimal *gbest* early. Computational resources are then freed on each iteration as low-scoring particles are selectively removed. The swarm size decay algorithm below uses a decay factor $0 < \gamma \ll 1$.

Algorithm 2: Swarm size decay algorithm

input : N_Q, $\mathcal{Q}$, *pbest*, γ, minimum swarm size N_{min}

output: N_Q, $\mathcal{Q}$

if $(\gamma > 0 \ \& \ N_Q > N_{min})$ **then**
  $N_Q \leftarrow \texttt{round}(1 - \gamma \times N_P)$;
  if $N_Q < N_{min}$ **then**
    $N_Q \leftarrow N_{min}$;
  $S_{pbest} \leftarrow \texttt{fitness}(pbest)$;
  $[\_, ID] \leftarrow \texttt{sort}(S_{pbest})$;
  $fittest \leftarrow ID(1 : N_Q)$;
  $\mathcal{Q} \leftarrow \mathcal{Q}(fittest)$;
return $N_Q, \mathcal{Q}$

3.3 Results and Discussion

3.3.1 Case Study Application

An example of a structured environment (i.e. an environment where the terrain, static obstacles, and environmental loading conditions are known or can be estimated with a high degree of confidence) are offshore wind farms like the Anholt Wind Turbine Array (figure 3.4), which we use as a case-study application.

Wind turbines require annual inspection of the submerged structure and power cables (DNVGL-ST-0126, 2018), which is normally completed using ROVs or divers. The distributed inspection mission aims to allocate visual inspection tasks to a fleet of REMORA AUVs, meaning we can use assumptions 1 and 2 for defining $\mathcal{M}_O$. Though the visual inspection of wind turbine substructure and cables is not as difficult a robotic control task as, for instance, underwater valve manipulation on offshore pipelines, the example stands as a proof-of-concept, multi-robot, task allocation and routing problem with variable sea conditions and known obstacles.

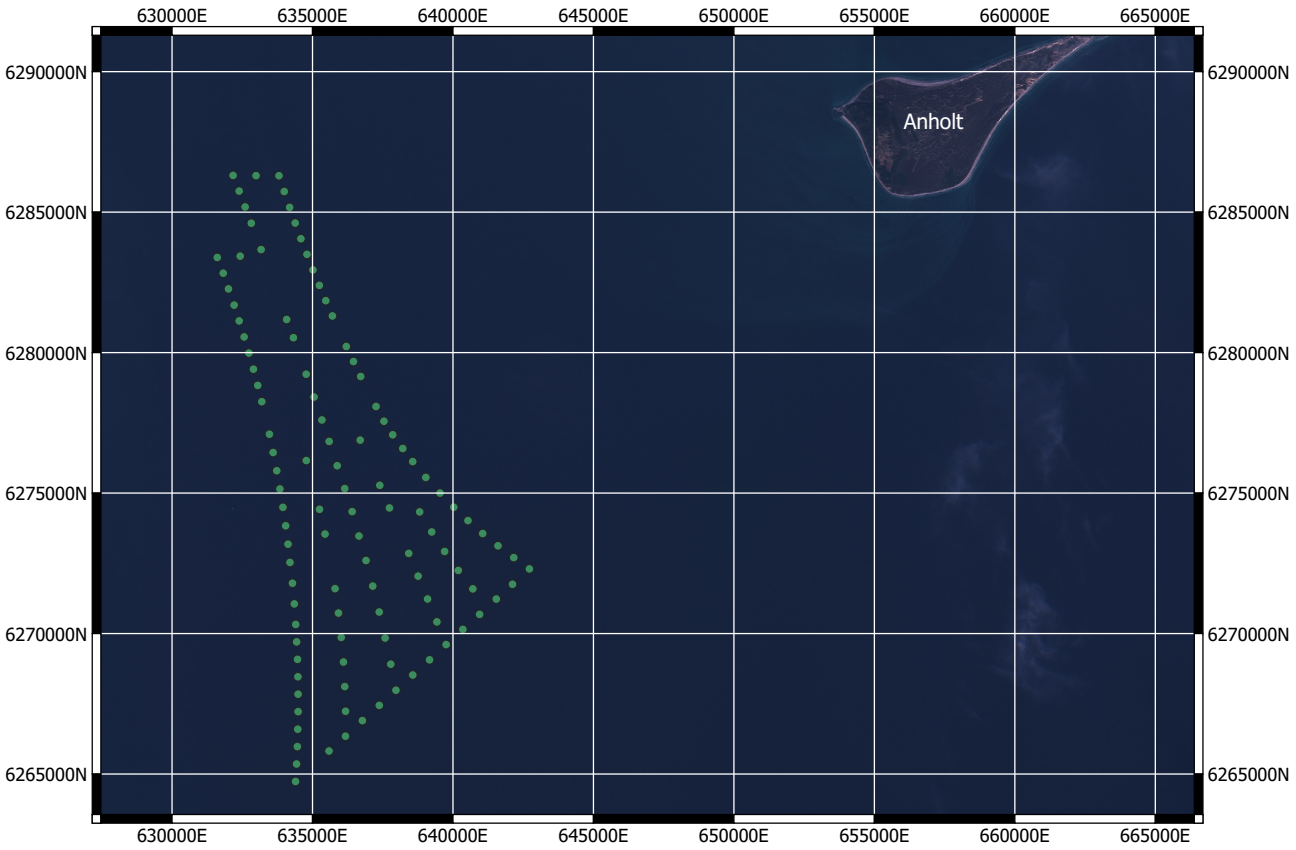


Figure 3.4: False colour map of Anholt array using infrared satellite imagery from Sentinel-2A (March, 2018) referenced to UTM zone 32N. Green dots indicate the captured centroid of each turbine.

Referring to the AMV mission planning definitions in section 3.2.2, the inspection mission needs to first be sub-divided into independent operating zones (defined by N_C instances of $\mathcal{M}_O$) which can then be formulated into the set of inspection tasks and special *home point* tasks, $\mathcal{T}$. The inspection task T is a helical trajectory the REMORA vehicle must follow to visually inspect the outer surface of a wind turbine substructure from a point close to the waterline to the seabed. All inspection tasks are given an equal reward $s = 1$, and g is set to be 6 m from the centroid of a wind turbine (maintaining a 1 m distance from the exterior of the turbine substructure). There are 111 wind turbines, meaning that $N_T \geq 113$ depending on the feasible operating zone clustering.

Three REMORA vehicles will be used for the inspection mission ($N_V = 3$). The REMORA vehicle can be configured with two 14.8V, 6.2Ah LiPo batteries connected in parallel (12.4Ah total capacity). Each vehicle's available energy capacity, e_b, is then calculated to be approximately 462 kJ from equation 3.10, with 30% of the full capacity kept as an emergency reserve. The parameters of the REMORA dynamic model from equation 3.5 have been empirically determined through model tests by Nielsen et al. (2018). For the homogeneous fleet assumption,

$\mathcal{V}$ has now been adequately defined.

3.3.2 Feasible Operating Zone Clustering

Now that the wind turbine inspection mission is sufficiently defined in terms of task location and vehicle constraint data, the first step of the mission planner procedure is to subdivide the mission area (i.e. the area encapsulated by the location data in $\mathcal{T}$) into feasible operating zones for the vehicles. With a constant forward velocity of $\bar{U} = 0.5$ m/s, the calculated PSR for the REMORA vehicle is 4660.5 m. The PSR is used in algorithm 1 along with the inspection task locations $\mathcal{T}_{2,\ldots,N_T-1}$, to obtain figure 3.5.

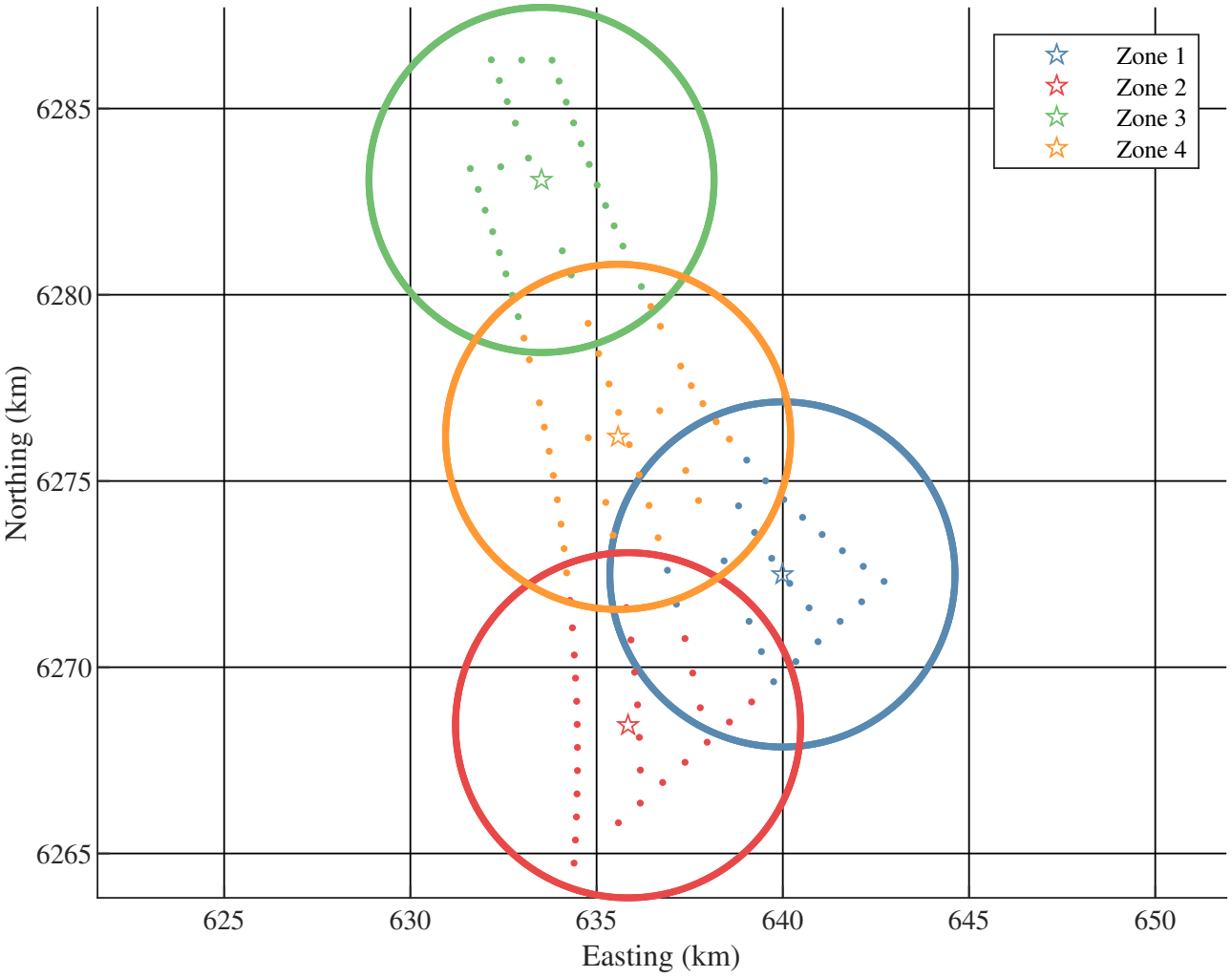


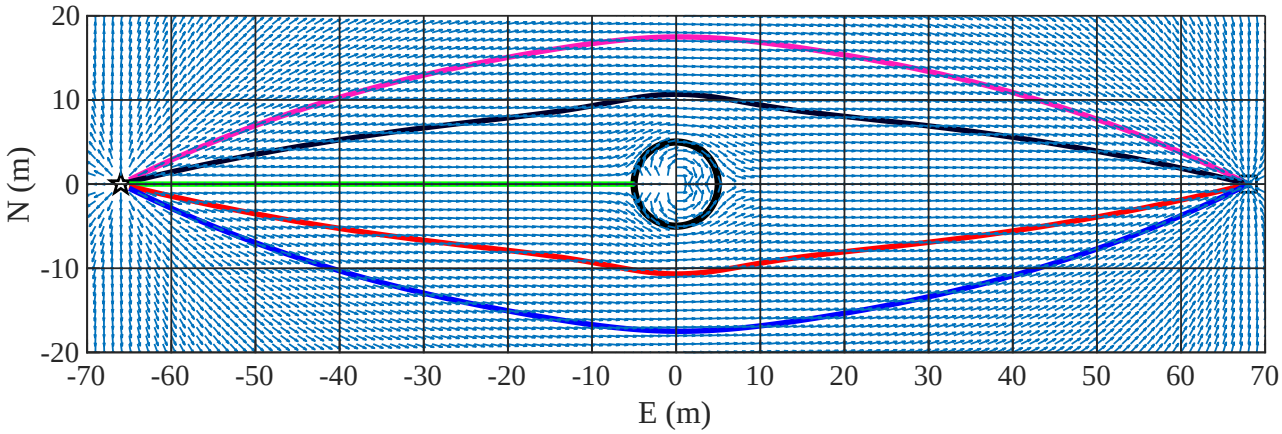
Figure 3.5: The Anholt wind turbine array, clustered according to the mean PSR of the REMORA vehicle (4.66 km). Each zone's star is the home point (the start/finish position for all vehicles) and centre of the respective cluster. Overlaps between each zone's PSR and another zone's target set presents the opportunity for inter-zone assistance.

The full $\mathcal{M}_O$ is then decomposed into N_C instances, where $\mathcal{T}$ is distributed to each new $\mathcal{M}_O$ according to the clustering algorithm. The full inspection mission is then formulated into

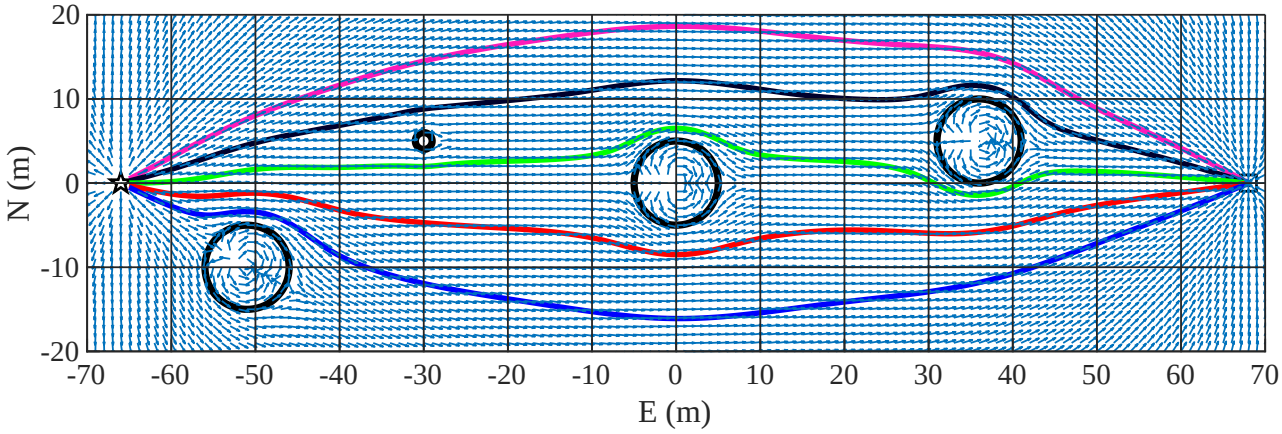
independent N_C sub-missions $\mathcal{M}_O^{(i)} \mid i \in \{1, \ldots, N_C\} \subset \mathbb{N}$. For each $\mathcal{M}_O^{(i)}$, the special *home point* task locations, $g \in \mathcal{T}_{\{1,N_T\}}^{(i)}$, are set to the location of the i-th cluster centroid. Each $\mathcal{M}_O^{(i)}$ can then be digested by the mission planner search algorithm (DStPSO) into a corresponding $\mathcal{M}_C^{(i)}$ for optimisation. But first, $\mathcal{S}$ must be generated for each possible transition in each $\mathcal{M}_O^{(i)}$, so that the corresponding $E^{(i)}$ matrix can be obtained.

3.3.3 Evaluation of Hydrodynamic Potential Flow Path Planning

Hydrodynamic potential flow presents an efficient solution to obtaining a path $\mathcal{S}$ that navigates around obstacles at a constant forward velocity, but it has two vulnerabilities. Stagnation points on the boundary of an obstacle that cause the vehicle to be trapped in a position of zero velocity, and the generated path having a curvature that cannot be adequately followed due to the vehicle manoeuvrability constraints. Figure 3.6a shows an example of the stagnation point causing the vehicle to get stuck in a local minima at the obstacle boundary. This scenario is only likely to happen when there is only one obstacle and its centroid lies on the line between the source and the sink. The influence from multiple obstacles (figure 3.6b), noise from the vehicle's location estimate, and the trajectory tracking error of the vehicle's controller all contribute in reducing the likelihood of the stagnation problem.



(a) Single obstacle.



(b) Multiple obstacles.

Figure 3.6: Example flow field and vehicle path (streamlines) using hydrodynamic potential flow. The green trajectory in 3.6a meets with a stagnation point on the surface of the obstacle located at (0,0). In practice this is unlikely to occur as the starting location must lie on the line between source and sink. The influence from multiple obstacles also reduces the likelihood of stagnation as in 3.6b.

Integrating the CV field (equation 3.18) from a starting point to a finishing point provides $\mathcal{S}$ based on a mass-less particle drifting along a streamline within the potential field. This ignores the inertial, hydrodynamic, and control components of the vehicle model (see equation 3.5). The vehicle dynamics may also cause a collision free trajectory to be invalid because the vehicle is unable to follow the path. This is due to the required turning rate, r, becoming too high for the vehicle's forward velocity, causing an error offset that the vehicle's controller cannot stabilise. This is likely to happen when the vehicle's trajectory is heading towards the centroid of an obstacle, requiring a large deflection around the obstacle by the integrated CV field. By artificially inflating the size of the obstacles, the radius of curvature of the generated path becomes larger, hence decreasing the magnitude of the required r. The size of the inflation can be determined either by iteratively increasing the inflation layer until the curvature of the solution satisfies the vehicle's r (or, if known, the vehicle's turning radius as in Murthy and

Rock (2010)).

Figure 3.7 shows a test trajectory generated for the REMORA vehicle model that must navigate around a circular obstacle to reach the position (17,0.1). $\mathcal{S}$ contains the attained x and y positions, commanded forward speed U, and commanded heading ψ for the integrated time series t. Figure 3.8 presents the commanded and achieved dynamics of the vehicle for the test trajectory when in autopilot and dynamic positioning modes, showing that the controller is able to adequately track the commands obtained from the integration of the CV field with the inflated obstacle.

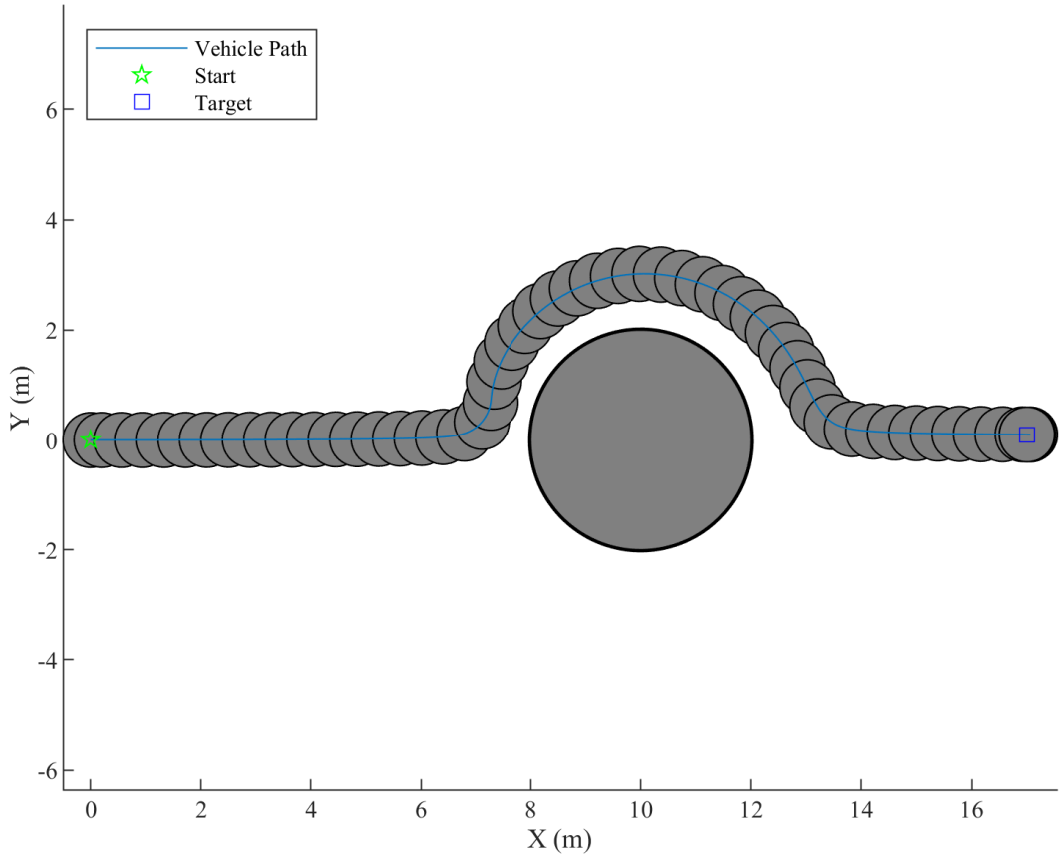


Figure 3.7: Path navigated by the REMORA Simulink model. Model started at (0,0) with orientation 0 radians (parallel to x axis) and was commanded to navigate to (17,0.1) using the CV flow equation. An obstacle, pictured at actual size, located at (10,0) with radius 5.0 m was inflated by 1.5 m (three times the vehicle's collision radius) for the CV field equations. The resulting path produces a trajectory with curvature suitable for the vehicle to track, avoiding collision with the actual obstacle.

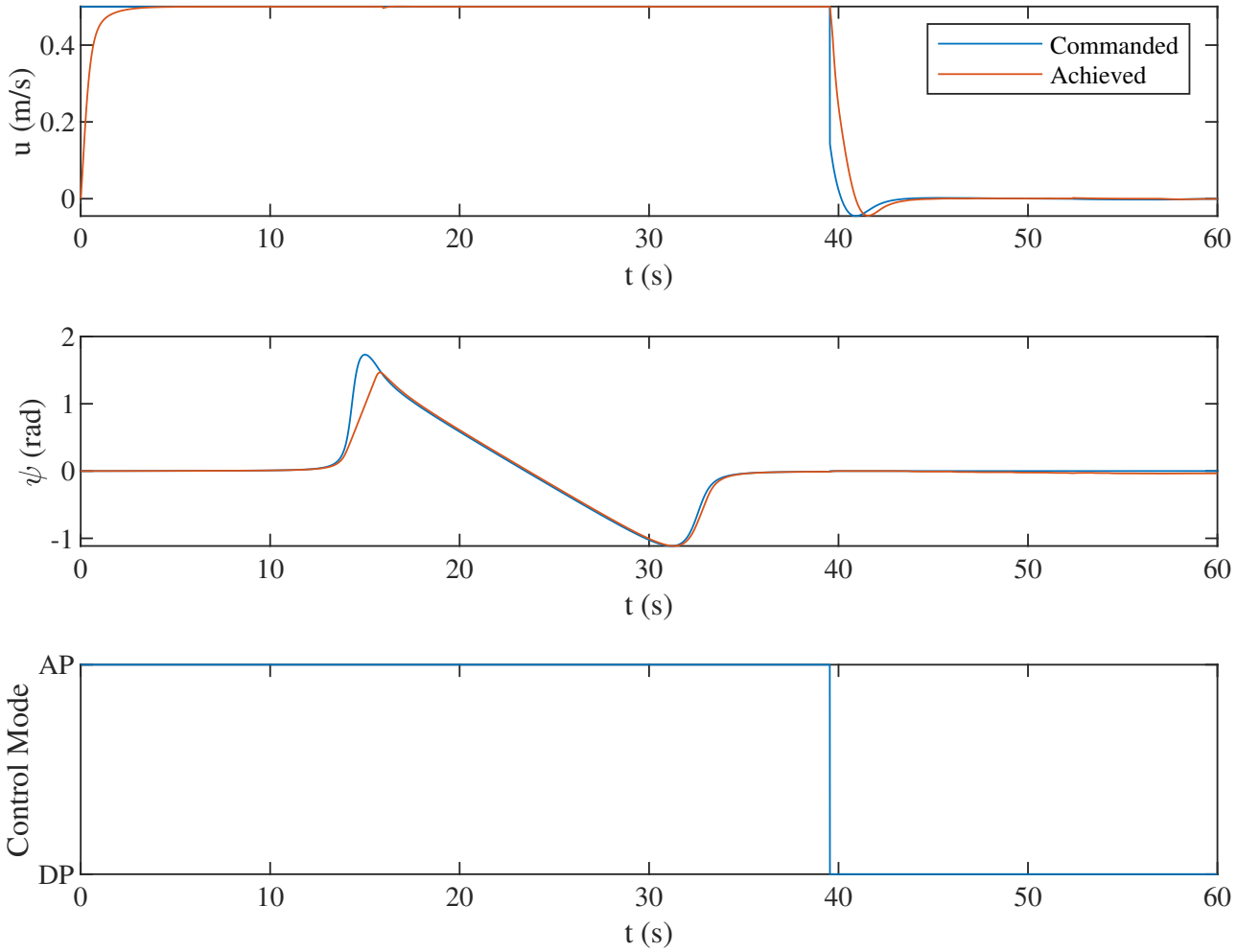


Figure 3.8: Top: Commanded and actual forward speed of the model during the transition. Middle: Commanded and actual heading of the model during the transition. Bottom: The control method switches between Autopilot (AP) and Dynamic Positioning (DP) mode when the vehicle gets within 0.2 m of the destination. DP enables high manoeuvrability and control but consumes more energy than AP.

3.3.4 Swarm Size Decay Evaluation

The DStPSO solver was modified in section 3.2.8 to include a linear swarm size decay algorithm that prunes the swarm of poor performing particles on each iteration. A performance comparison between the original DStPSO ($\gamma = 0$) and the decayed DStPSO ($\gamma > 0$) was made using the three location data sets from Tsiligirides (1984). The three data sets have been formulated into 147 test problems that vary in the maximum time constraint and number of vehicles, available from KU Leuven (2018). Data sets from Tsiligirides (1984) were chosen because they are of similar problem space complexity to the clustered Anholt array.

The DStPSO algorithm was initialised with inertia weight $w = 0.7$, social bias weight $c_1 = 0.5$, self bias weight $c_2 = 0.5$. Stopping criteria is achieved after 300 consecutive iterations of no improvement (stall). RVNS was set to move from neighbourhood 2 to neighbourhood 3 after

10 consecutive iterations of no improvement, and stopping criteria was set to trigger after 20 consecutive iterations of no improvement from neighbourhood 3. Three solver configurations (varying in γ) were trialled over 10 repeats, measuring the computational time (CPU), Relative Percentage Error (RPE) from the best solution of a particular test problem, and the standard deviation of the accumulated reward (σ). The solver was implemented in MATLAB and tested on an Intel i5 2.3 GHz PC with 1 GB of memory. Figure 3.9 presents the results averaged over the entire test set.

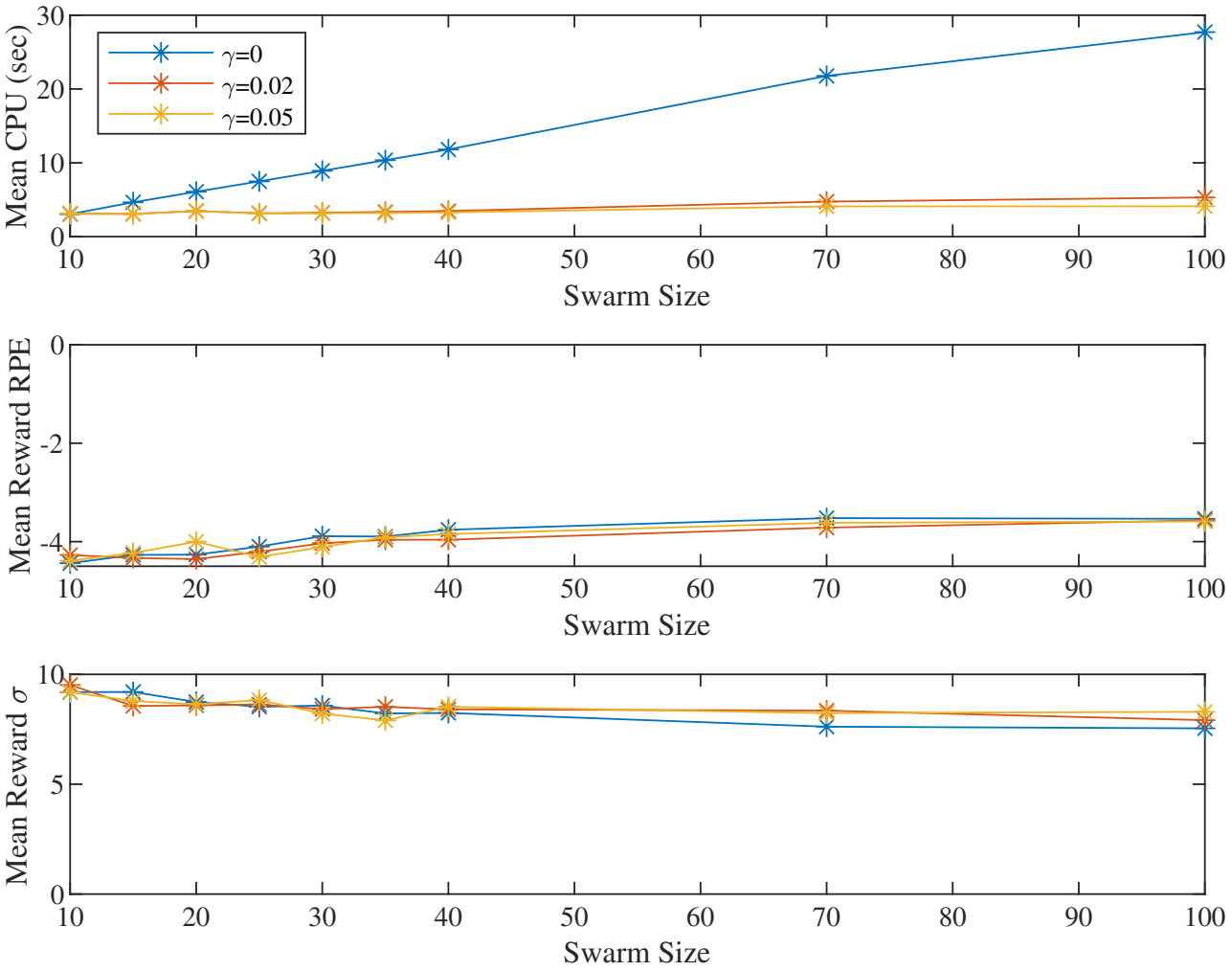


Figure 3.9: Performance comparison of modified DStPSO algorithm with varying γ. CPU time, reward RPE, and reward σ are averaged over all 147 test sets, repeated 10 times each.

Comparison of RPE and reward σ across each γ variant shows that larger swarm sizes converge towards a common optimum (σ decreases and RPE increases). The major performance advantage is observed in CPU time difference. For the same swarm size, DStPSO with $\gamma = 2\%$ is near 10 times faster than the original DStPSO. The original DStPSO with a swarm size of 10 has a similar CPU time to the 2% DStPSO of swarm size 100, but has a lower RPE and a higher σ. Additionally, there is a higher amount of variance in the smaller swarm sizes. Increasing the

amount of repetitions for each solver configurations should produce a clearer trend.

3.3.5 Optimisation of Mission Proposal

As determined in section 3.3.2, there are four sub-missions that must be solved by the DStPSO algorithm in order to provide a complete plan for the inspection mission of the Anholt array. As shown in figure 3.1, the trajectory generator requires knowledge of the static obstacles' positions and radii (which are provided from the obstacle database as the tuple $O = (X_o, r_o, I_o)$) and the start and end points for the trajectory ($g \in \mathcal{T}_{\{i,j\}}$) for it to produce the requested trajectory $\mathcal{S}_{ij}$.

For the inspection mission case study, each turbine substructure is a pile 5 m in radius, whose cross-section can be represented on the East-North (X-Y) plane as circles of 5 m radius. $\mathcal{O}$ is then the collection of 111 X_o coordinates of each turbine location, and r_o is the collection of the corresponding 111 substructure radii, which are all set to 5 m. Given the starting (i-th) and finishing (j-th) coordinates from the k-th sub-mission proposal, $\{g \in \mathcal{T}_{\{i,j\}} | \mathcal{M}_C^{(k)}, \mathcal{O}^{(k)}\}$, the trajectory generator can produce $\mathcal{S}_{ij}$ for each sub-mission.

Each $\mathcal{S}_{ij}$ produced by the trajectory generator is mapped to the corresponding element E_{ij} using the method in section 3.2.3. $\mathcal{S}_{ij}$ provides the time interval over which $E_{s,ij}$ and E_h are obtained. $E_{t,j}$ is obtained from the nested energy consumption prediction of the helical inspection task, which will have a different $\mathcal{H}$ and $\mathcal{P}$ from the transition phase because special inspection equipment (cameras, sonar, etc.) will be active at this point in the task, and the 3D trajectory taken by the vehicle around the substructure is significantly different from the planar transition trajectory. For the sake of brevity, we have assigned the expected task energy consumption $E_{t,j} = 1\,\mathrm{kJ}\,\forall E_{t,j} \in E$, meaning that a constant is depleted from the vehicle's battery for every task it completes.

Having obtained $E \in \mathcal{M}_O^{(k)}$, the DStPSO algorithm is used to evaluate an optimum $\mathcal{M}_C^{(k)*}$ as described in section 3.2.8. The final *gbest* corresponds to $R \in \mathcal{M}_C^{(k)*}$. The route for the vehicle $l \in Q$, R_l can then be used to access the set of trajectories $\{\mathcal{S}_{ij} \mid (i,j) \in R_l, 1 \leq i \leq L_l - 1, 2 \leq j \leq L_l\}$. We present the set of trajectories proposed by $\mathcal{M}_C^{(2)}$ in figure 3.10.

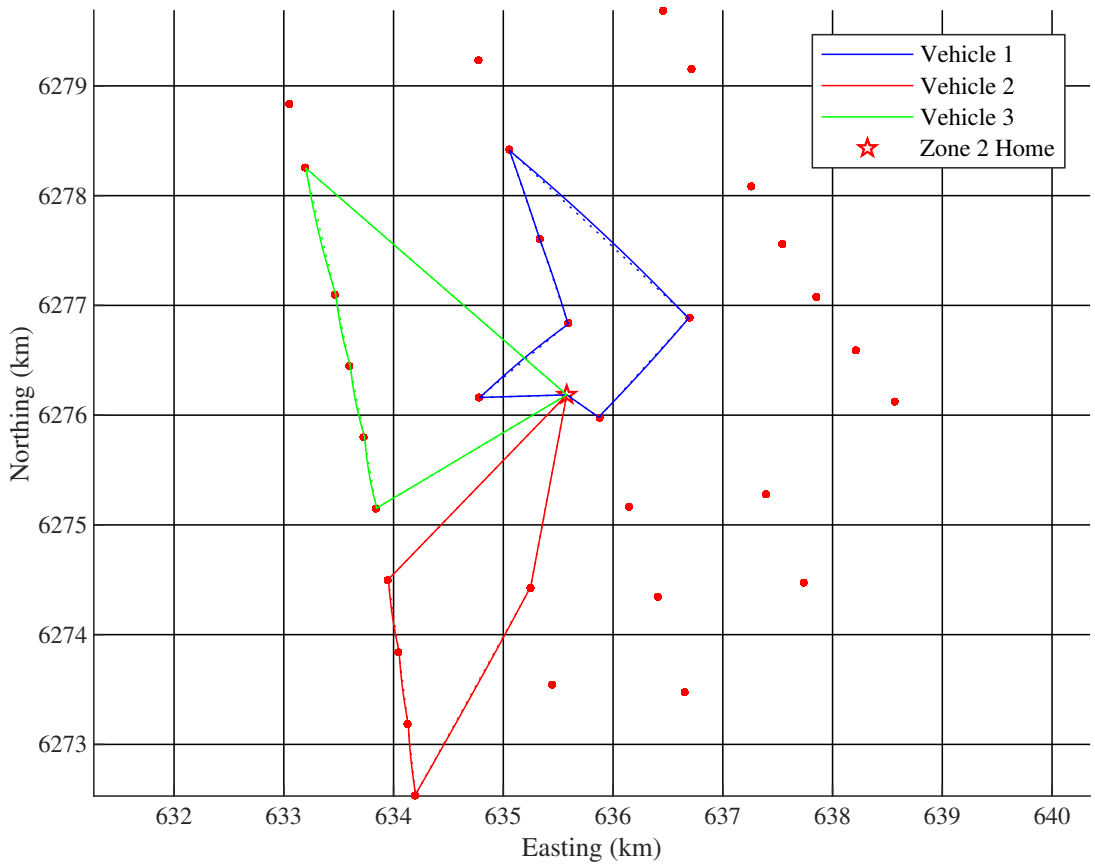


Figure 3.10: Optimised route for cluster zone 2. Due to minimum line thickness, obstacles are not to scale and are larger than actual. Dotted lines represent the Euclidean path of the corresponding vehicle route. The star denotes the home position (start/finish point for the vehicles). As is the case with most of the transitions, the shortest path was a straight line path to the destination, with small distortions in the path that flow around obstacles.

3.4 Conclusions

3.4.1 Summary

A new mission planner framework for AMVs was proposed, formulated as the TOP from operations research. The mission domain was first defined in its open form, containing information about the tasks, vehicles, and obstacles as specified by the AMV operator, the *knowledge based reasoning* step discussed in section 3.1. The mission planner searches through the open mission $\mathcal{M}_O$ for an optimum proposal, called the closed mission $\mathcal{M}_C$. The closed mission is an initial plan that contains task allocation and sequencing information for the AMV fleet to execute. Here it can be seen that the *task allocation* step of mission planning has been completed.

The planner differs from temporal planners and task hierarchy planners because it uses energy

as the base finite resource. Considering energy means considering the loading on a particular vehicle over the extent of its mission. It requires a trajectory generator to produce viable paths that can be assessed for energy consumption using the dynamic model of the vehicle under consideration. Treating energy consumption as the expected variable of a stochastic process means that uncertainty has been considered by the planner. This is the foundation of the third step in mission planning, *risk projection*. In future development of the mission planner, the level of allowable uncertainty in the mission plan can be specified by the operator as a constraint.

The mission planner is modular in nature because the definition of the open mission requires several separate databases to be processed into the open mission formulation. This means that the components specified in section 3.2 are interchangeable with different or more advanced methods, depending on the complexity of the mission.

The integration of the components into the mission planner framework also produced 'spillover effects' as minor improvements to the literature concerning some of the components. Most notable is the improvement of the DStPSO algorithm with the swarm decay modification. It is shown in section 3.3.4 that the modification allows for a wider initial exploration of the search space with more particles whilst saving computational resources in the later stages of the search. The usefulness of swarm decay depends on the complexity of the problem space. Discrete problems such as the TOP are suitable. However continuous domain problems, and planning problems with several layers of dependency may not be adequately explored if γ is set too high.

As was shown in section 3.3.3, the hydrodynamic APF trajectory generator could be used with the REMORA vehicle's dynamic model to generate a collision free curve, but is open to the risk of being caught in the stagnation point of the an obstacle boundary. We therefore recommend the hydrodynamic APF method used by Waydo and Murray (2003), Owen et al. (2012) and adapted with marine vehicle specialised controllers such as in Pedersen and Fossen (2012), as a path planning implementation for the guidance of marine vehicles.

Finally, we tested the mission planner with simulated operator input data from case-study inspection mission of the Anholt wind turbine array. Following the homogeneous fleet and non-hierarchical task assumptions stated in section 3.2.2, we formulate the test data into four separate open missions using the feasible operating zone component. DStPSO was then shown to successfully obtain the closed mission proposal for each instance.

3.4.2 Future Work

The proposed AMV mission planner stands as the preliminary framework for more generalised AMV missions. The current solver, capable of solving TOP type missions, only works for the special homogeneous fleet, non-hierarchical task case. The planner can be extended to include heterogeneous fleets of vehicles in three ways:

1. Defining E as the matrix $\mathbb{R}^{N_T \times N_T \times N_V}$ allows for different vehicle dynamic models to be used according to $I_v \in V$.
2. Extending P as a N_V long set of control vectors that reference $\{T_{P_i} \in \mathcal{T} \mid 1 \leq i \leq N_V, i \in \mathbb{N}\}$, essentially defining the set of tasks in $\mathcal{M}_O$ that each vehicle is capable of doing according to $I_v \in V$.
3. Modifying $s \in S$ by a scalar utility variable found in $I_v \in V$ allows for vehicles more effective at completing certain task types than others to evaluate as a higher scoring solution than alternative solutions.

Additionally, the planner can be extended to include hierarchical tasks (i.e. tasks that depend on the completion of other tasks) by specifying a prerequisite variable in $I_t \in T$. This allows for a logical hierarchy, but then must be further extended using temporal logic in order to obtain an energy efficient hierarchical proposal.

3.4.3 Considerations for Energy Planning During Deployment

Referring to figure 3.1, the energy planner's primary function is to perform the sequencing and allocation of tasks to vehicles within a fleet, meaning that it is functionally similar to the 'behaviour allocation' component in figure 2.3. Instead of working with behaviours (also known as actions), the planner formulates the open mission, $\mathcal{M}_\mathcal{O}$ (see definition 4), on the task level of abstraction. The $\mathcal{M}_\mathcal{O}$ representation is an adaptation of the TOP for the marine vehicle planning domain. The TOP solver's objective is to maximise the reward from completed tasks while conforming to energy and specific task sequencing constraints.

Because the planner does not perform true decomposition, the onus is on the operator to compose (i.e. reverse of the decomposition process; abstraction) available vehicle behaviours into available tasks and decompose the mission goals into required tasks. The intersection between the required and available tasks are then made available to the planner. In the offshore windfarm dataset (section 3.3.1) there are three types of tasks: starting from the deployment

zone, performing an inspection of a turbine substructure, and finishing at the rendezvous zone. Both the start and finish tasks are essentially the same in that they simply need the vehicle to move from one point to another. The sequence of behaviours required to complete the inspection task and that also belong to the set of available vehicle behaviours are:

- Move from current position to the turbine region.
- Perform substructure inspection trajectory and record payload.

These behaviours are bundled into the hierarchical task representation tuple (see definition 1) for each task, which is a component of $\mathcal{M}_{\mathcal{O}}$. The planner finds a solution (the closed mission $\mathcal{M}_{\mathcal{C}}$ from definition 5) by searching through $\mathcal{M}_{\mathcal{O}}$ using an algorithm designed to solve the TOP.

3.4.4 Challenges for Energy Planning

In section 2.6.2, temporal-based AI and OR planning approaches were briefly discussed and compared. One of the properties of temporal planning is that the calculation of the time resource for a vehicle need only be performed at a kinematic level. For example, the time for a vehicle to transition from point A to point B can be estimated based on its operating speed and the length of the expected trajectory. In comparison, energy planning requires consideration of the forces that cause the desired motion of the vehicle as it is these forces that allow the estimation of the mechanical work required. In this respect, the information used by the energy planner will more closely represent the physical motion of the vehicle than a temporal approach. The process by which this information is obtained involves more steps and uses more variables (input data) than the temporal approach (figure 3.11). This increases the computational load (and solving time) of the planner and increases the risk of producing an unrealistic plan due to inaccurately identified variables.

A dynamic model for the vehicle based on accurate system identification of the hydrodynamic and inertial parameters must be used in conjunction with an environment model to calculate the body forces required to follow a reference trajectory. The forces are then decomposed into the thrusts and moments that need to be output by individual actuators on the vehicle, preferably according to the control allocation process that is used by the vehicle. The mechanical power required by each actuator to exert their allocated forces needs to be estimated through specific models of each actuator type (such as a thrust-RPM curve for a propeller or pitching moment-angle of attack curve for a hydroplane). Mechanical and electrical efficiencies of the actuator drivetrain need to be identified to obtain the actuator power requirements. The hotel

load for the rest of the system needs to be included to approximate the total system power requirements. Finally, a model of the battery's internal resistance must be considered for an accurate prediction on the power draw across the battery terminals.

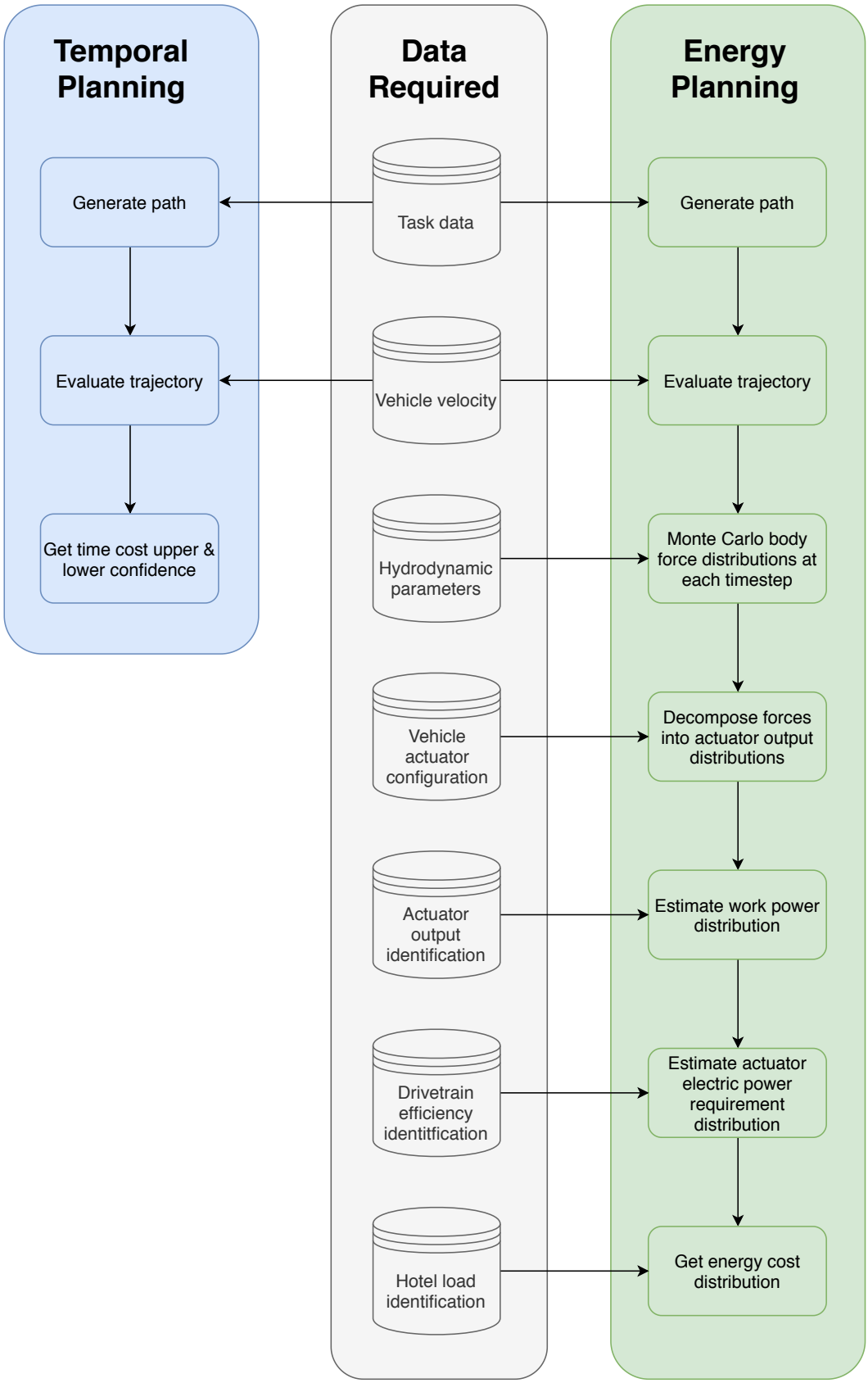


Figure 3.11: Comparison of knowledge base creation process between temporal and energy planning. Energy planning has significantly more steps and data variables but makes a more informed prediction of the vehicle's performance in the environment.

3.4.5 Representing Uncertainty

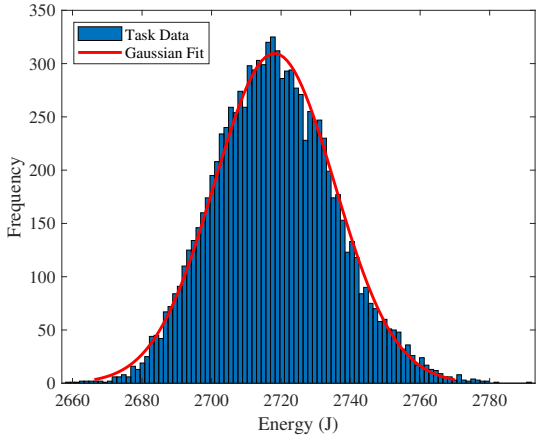
Uncertainty in the time cost prediction of temporal planning is accounted for by specifying upper and lower speeds for the vehicle, thus giving upper and lower time estimates for the temporal planner. In the energy approach proposed in section 3.2.3, the vehicle hydrodynamic model parameters that influence the energy cost prediction were converted into random variables by adding white noise scaled to the variance of the model identification experiment results (Nielsen et al., 2018). Monte Carlo simulation was then used to obtain a distribution of the expected energy costs for a trajectory. Depending on the sample size of the simulation and the length of the projected trajectory, the calculation of energy cost distributions for each task combination can become quite taxing on the planner, but only needs to be performed for the full set once during the creation of $\mathcal{M}_{\mathcal{O}}$. The advantage of sampling the model as random variables means that the effects of uncertainty in the model parameters are included into the energy distribution.

To reduce the computational loads and communication overhead required for the planner to communicate energy distribution information to vehicles, it would be convenient to approximate them as a standard distribution fit. The classification of the fit depends upon the models used in the simulation. The hydrodynamic model used in section 3.2.3 does not implement non-linear terms, which results in task energy requirement distributions output by the Monte Carlo simulation that are Gaussian-like in appearance (see figures 3.12a and 3.12b).

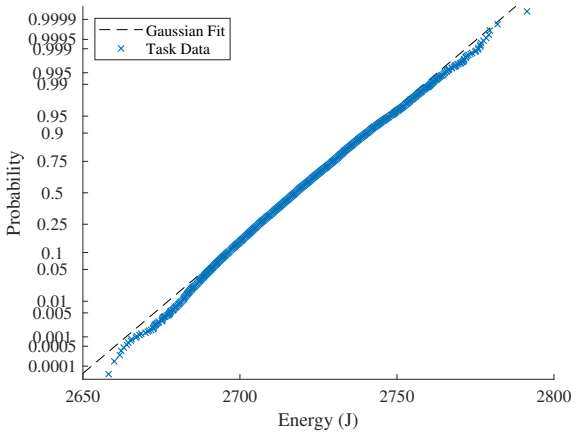
These distributions were tested for normality using the Anderson-Darling test, and were found to not be Gaussian at a significance level of 5%. The Gaussian probability plots of the data (figures 3.12b and 3.12d) show that approximating the distributions as a Gaussian distribution will result in approximation errors at the upper and lower 5% boundaries. Furthermore, the Gaussian approximation is likely to overestimate the likelihood of the energy distribution at the lower 5% boundary and is likely to underestimate the likelihood of the energy distrubtion at the upper 5% boundary, making the approximation optimistic. The principle benefit of using a Gaussian approximation is that each task distribution can be represented with just two parameters (mean and variance), minimising the amount of data that needs to be transmitted between the planner and the vehicle and minimising the computational resources required to perform planning operations with the distributions. Given the above consideration of the trade-off between optimism bias error and practicality, the remainder of this and the next chapter operate under the assumption that a Gaussian approximation adequately represents

the uncertainty of the underlying hydrodynamic model[2].

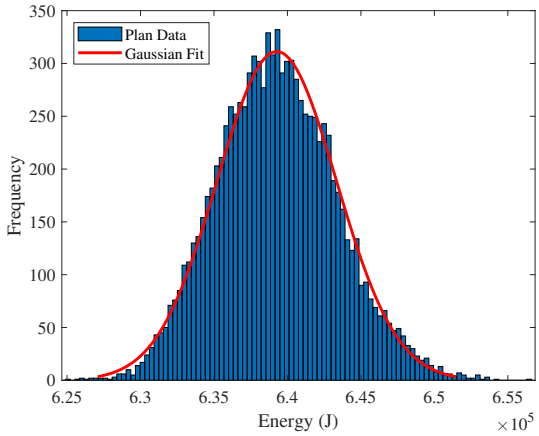
Following the rule that a weighted sum of Gaussian distributions results in a Gaussian distribution, the predicted energy consumption distribution for the entire plan (which is a summation of tasks) is also Gaussian (figures 3.12c and 3.12d). For models that include the higher order non-linear terms or cross-coupled terms, the resulting distribution is unlikely to adequately fit a Gaussian distribution approximation.



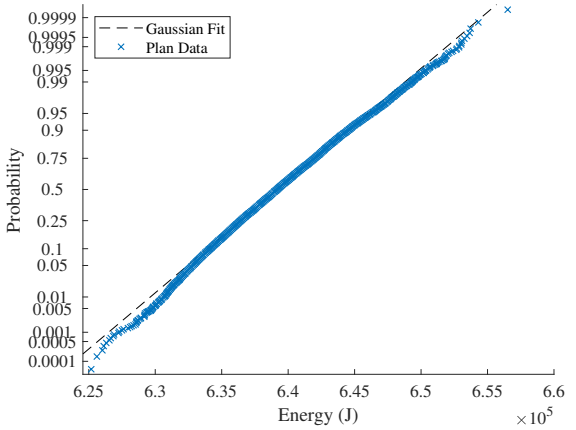
(a) Task distribution with Gaussian fit.



(b) Task probability plot with Gaussian fit.



(c) Plan distribution with Gaussian fit.



(d) Plan probability plot with Gaussian fit.

Figure 3.12: Example task and plan energy distribution plots. Both task (3.12a & 3.12b) and plan (3.12c & 3.12d) distributions are compared with a Gaussian fit.

3.4.6 Planning with Risk and Uncertainty

The previous section identified that the effects of uncertainty in the accuracy of the vehicle dynamics model on the energy prediction for a task or plan produce distributions that, for

[2]It's amusing to acknowledge that this assumption could be a sign of the planning fallacy in action (Kahneman and Tversky, 1977)

practical reasons, can be approximated as Gaussian distributions. The energy planner in section 3.2.3 only uses the expected value of the task distributions during the formulation of the energy consumption matrix E (see definition 4) while searching for $\mathcal{M}_{\mathcal{C}}$. This approach is pragmatic but disregards the rest of the distribution, meaning that the planner works with deterministic constraints. To accommodate the energy distribution, the planner could be expanded into a stochastic constraint optimisation problem (an optimisation problem with some constraints that are random variables) such as a team variant of the Orienteering Problem with Stochastic Weights (OP-SW) (Evers et al., 2014, Shang et al., 2016).

So far, the OP-SW has been approached by first solving it according to the expected costs of the edges in the OP graph. Then a second stage of the solver is invoked while the vehicle is deployed and is realising the true cost of the edge. A 'return home' recourse action is invoked by the solver to minimise the risk of violating the total allowable cost constraint. A crucial assumption is that the vehicle has an 'emergency reserve' budget not included by the solver so that the 'return home' action is successful regardless of the realised cost of returning home. In this respect, the vehicle could benefit from having alternative recourse actions that depend on other criteria besides the total cost constraint, resulting in opportunistic reward seeking strategies or in conservative survival-seeking strategies.

Given the communication challenges of the marine environment (section 2.5.3), there is a need for the vehicle to perform the second stage solving process *in situ*. Referring once again to figure 2.3, this second stage solver fulfils the 'execution monitor' component. The next chapter will present developments to the energy planner that enable it to operate with stochastic weights, investigate alternative recourse actions that promote opportunistic or risk-averse strategies, and provide a risk-based metric that the operator can use to specify the policies of the vehicle prior to deployment.

References

Ab Wahab, M. N., Nefti-Meziani, S., and Atyabi, A. (2015). A comprehensive review of swarm optimization algorithms. *PLOS ONE*, 10(5):1–36.

Ai-Chang, M., Bresina, J., Charest, L., Chase, A., Hsu, J. C. J., Jonsson, A., Kanefsky, B., Morris, P., Rajan, K., Yglesias, J., Chafin, B. G., Dias, W. C., and Maldague, P. F. (2004). MAPGEN: mixed-initiative planning and scheduling for the Mars Exploration Rover mission. *IEEE Intelligent Systems*, 19(1):8–12.

Brooks, R. A. and Lozano-Pérez, T. (1985). A subdivision algorithm in configuration space for findpath with rotation. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(2):224–233.

Chao, I., Golden, B. L., and Wasil, E. A. (1996). The team orienteering problem. *European Journal of Operational Research*, 88(3):464–474.

Choi, S., Lee, J. Y., and Yu, W. (2010). Fast any-angle path planning on grid maps with non-collision pruning. In *2010 IEEE International Conference on Robotics and Biomimetics*, pages 1051–1056.

Croes, G. A. (1958). A method for solving traveling-salesman problems. *Operations Research*, 6(6):791–812.

Dantzig, G. B. and Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, 6(1):80–91.

DNVGL-ST-0126 (2018). Support structures for wind turbines. Standard, Det Norske Veritas and Germanischer Lloyd, Oslo, Norway.

Evers, L., Glorie, K., van der Ster, S., Barros, A. I., and Monsuur, H. (2014). A two-stage approach to the orienteering problem with stochastic weights. *Computers & Operations Research*, 43:248–260.

Furlong, M. E., Paxton, D., Stevenson, P., Pebody, M., McPhail, S. D., and Perrett, J. (2012). Autosub long range: A long range deep diving auv for ocean monitoring. In *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, pages 1–7.

Furno, L., Blanke, M., Galeazzi, R., and Christensen, D. J. (2017). Self-reconfiguration of modular underwater robots using an energy heuristic. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6277–6284.

Gerkey, B. P. and Matarić, M. J. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *The International Journal of Robotics Research*, 23(9):939–954.

Giger, G. F. (2010). *An Operator-centric Mission Planning Environment to Reduce Mission Complexity for Heterogeneous Unmanned Systems*. PhD thesis, Pennsylvania State University, University Park, PA, USA. AAI3436143.

Gunawan, A., Lau, H. C., and Vansteenwegen, P. (2016). Orienteering problem: A survey of recent variants, solution approaches and applications. *European Journal of Operational Research*, 255(2):315 – 332.

Hansen, P., Mladenović, N., and Moreno Pérez, J. A. (2010). Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175(1):367–407.

Hobson, B. W., Bellingham, J. G., Kieft, B., McEwen, R., Godin, M., and Zhang, Y. (2012). Tethys-class long range AUVs - extending the endurance of propeller-driven cruising AUVs from days to weeks. In *2012 IEEE/OES Autonomous Underwater Vehicles (AUV)*, pages 1–8.

Kahneman, D. and Tversky, A. (1977). Intuitive prediction: Biases and corrective procedures. Technical report, Defense Advanced Research Projects Agency. N00014-76-C-0074.

Kavraki, L. E., Svestka, P., Latombe, J. C., and Overmars, M. H. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580.

Kellerer, H., Pferschy, U., and Pisinger, D. (2004). *Knapsack Problems*, chapter 1, pages 1–14. Springer Berlin Heidelberg, Berlin, Heidelberg.

Khatib, O. (1985). Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, volume 2, pages 500–505.

KU Leuven (2018). The orienteering problem: Test instances. KU Leuven, https://www.mech.kuleuven.be/en/cib/op. Accessed: 2018-22-05.

Kuffner, J. J. and LaValle, S. M. (2000). RRT-connect: An efficient approach to single-query path planning. In *Proceedings of ICRA 2000: IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 2, pages 995–1001 vol.2.

Labadie, N., Mansini, R., Melechovský, J., and Calvo, R. W. (2012). The team orienteering problem with time windows: An lp-based granular variable neighborhood search. *European Journal of Operational Research*, 220(1):15 – 27.

Lesire, C., Infantes, G., Gateau, T., and Barbier, M. (2016). A distributed architecture for supervision of autonomous multi-robot missions. *Autonomous Robots*, 40(7):1343–1362.

Li, J. (2011). Model and algorithm for time-dependent team orienteering problem. In Lin, S. and Huang, X., editors, *International Conference on Computer Education, Simulation and Modeling*, pages 1–7, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mac, T. T., Copot, C., Tran, D. T., and Keyser, R. D. (2016). Heuristic approaches in robot path planning: A survey. *Robotics and Autonomous Systems*, 86:13 – 28.

McGann, C., Py, F., Rajan, K., Thomas, H., Henthorn, R., and McEwen, R. S. (2008). A deliberative architecture for AUV control. In *2008 IEEE International Conference on Robotics and Automation*, pages 1049–1054, Pasadena, CA, USA.

Mester, D., Bräysy, O., and Dullaert, W. (2007). A multi-parametric evolution strategies algorithm for vehicle routing problems. *Expert Systems with Applications*, 32(2):508 – 517.

Milne-Thomson, L. (2013). *Theoretical Hydrodynamics*. Dover Books on Physics. Dover Publications, Mineola, NY, USA.

Murphy, R. R., Steimle, E., Griffin, C., Cullins, C., Hall, M., and Pratt, K. (2008). Cooperative use of unmanned sea surface and micro aerial vehicles at Hurricane Wilma. *Journal of Field Robotics*, 25(3):164–180.

Murthy, K. and Rock, S. M. (2010). Spline-based Trajectory Planning Techniques for Benthic AUV Operations. In *Proceedings of IEEE Autonomous Underwater Vehicles Conference*, Monterey, CA, USA.

Nielsen, M. C., Eidsvik, O. A., Blanke, M., and Schjølberg, I. (2018). Constrained multi-body dynamics for modular underwater robots — theory and experiments. *Ocean Engineering*, 149:358–372.

Owen, T., Hillier, R., and Lau, D. (2012). Smooth path planning around elliptical obstacles using potential flow for non-holonomic robots. In Röfer, T., Mayer, N. M., Savage, J., and Saranlı, U., editors, *RoboCup 2011: Robot Soccer World Cup XV*, pages 329–340, Berlin, Heidelberg. Springer Berlin Heidelberg.

Pedersen, M. D. and Fossen, T. I. (2012). Marine vessel path planning & guidance using potential flow. *IFAC Proceedings Volumes*, 45(27):188 – 193.

Py, F., Pinto, J., Silva, M. A., Johansen, T. A., Sousa, J., and Rajan, K. (2016). EUROPtus: A mixed-initiative controller for multi-vehicle oceanographic field experiments. In *International Symposium on Experimental Robotics*, pages 323–340, Tokyo, Japan.

Richards, F. J. (1959). A flexible growth function for empirical use. *Journal of Experimental Botany*, 10(2):290–301.

Sevkli, Z. and Sevilgen, F. E. (2010). Discrete particle swarm optimization for the orienteering problem. In *IEEE Congress on Evolutionary Computation*, pages 1–8, Barcelona, Spain.

Sevkli, Z. and Sevilgen, F. E. (2012). Discrete particle swarm optimization for the team orienteering problem. *Turkish Journal for Electrical Engineering and Computer Sciences*, 20(2):231–239.

Shang, K., Chan, F. T. S., Karungaru, S., Terada, K., Feng, Z., and Ke, L. (2016). Two-stage robust optimization for orienteering problem with stochastic weights. *arXiv e-prints*, page arXiv:1701.00090.

Sotzing, C. C., Evans, J., and Lane, D. M. (2007). A multi-agent architecture to increase coordination efficiency in multi-AUV operations. In *OCEANS 2007 - Europe*, pages 1–6, Aberdeen, Scotland, UK.

Sotzing, C. C. and Lane, D. M. (2010). Improving the coordination efficiency of limited-communication multi–autonomus underwater vehicle operations using a multiagent architecture. *Journal of Field Robotics*, 27(4):412–429.

Tsiligirides, T. (1984). Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35(9):797–809.

Waydo, S. and Murray, R. M. (2003). Vehicle motion planning using stream functions. In *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, volume 2, pages 2484–2491 vol.2, Taipei, Taiwan.

Chapter 4

Field Trials of an Energy Aware Mission Planner Implemented on an Autonomous Surface Vehicle[1]

4.1 Introduction

This chapter considers automated mission energy planning and forecasting for AMVs with electric power supplies. The energy supply of an AMV is vital to its successful operation, power failures are one of the leading causes of overall mission failure (Brito et al., 2014). Effectively managing the energy supply means not only ensuring that the AMV does not overextend itself with an overambitious mission plan, it also means making the most use of the available energy of the vehicle to achieve as many mission tasks as possible. Currently, operators balance the survival of their vehicle and its task completion effectiveness by consulting the manufacturer specifications (e.g. range, endurance, rated depth) and applying safety factors to those specifications in order to obtain safe mission planning constraints. In pursuit of refining this balance to maximise the effectiveness of the vehicle while still maintaining survivability, this paper considers energy management from an onboard mission planning and decision-making approach.

The reasoning and deliberation capabilities of AMVs have blossomed over the last decade. Prior to 2008, mission planning for AMVs was a task reserved for the human operator, who

[1]This chapter is a reproduction of the article submitted to the Journal of Field Robotics on the 22nd February, 2019 and is currently under review.

would create a scripted set of way-points, depths, speeds, control modes, and sensor payload modes. This procedure relied on the operator's experience to predict the reliability of the mission script, and to adequately prepare the vehicle with contingencies for outcomes that are likely to occur. McGann et al. (2008) ushered in a new method of planning for AMVs by implementing the deliberative T-REX agent onboard AUVs. T-REX effectively allows vehicles to adjust their plan and behaviours during execution according to detected external changes and inferred changes to the mission state.

With AMVs able to adapt plans in order to achieve mission objectives, automated mission planning has been extended to dynamically generate and adapt mission plans for large operations. Mesoscale (≥ 50 km^2) coordinated multi-AMV operations have been realised through the development and implementation of temporal planners such as EUROPtus (Py et al., 2016). Temporal plans schedule and allocate tasks to the vehicles using time as the base resource constraint. A partial plan is instantiated and is refined into a complete plan as flaws are observed during execution. Temporal planning allows for easy synchronisation of individual vehicle plans, which is convenient for operators when deploying and retrieving vehicles (Ferreira et al., 2018), or for mixed-initiative missions (Ai-Chang et al., 2004). However, the environmental loadings experienced by the vehicles while deployed are not directly considered in temporal planning. Instead the planner relies on the time taken for the vehicle to perform tasks and its speed as the relevant temporal indicators.

In Thompson and Galeazzi (2018), an energy based planner was proposed to predict the energy cost for a team of vehicles to perform tasks. It then uses these predictions to schedule and allocate tasks to individual vehicles based on their available energy resources. Energy planning factors in the loadings on the vehicles traversing waypoints along an expected path (something that is not considered by temporal planners) and can be compared against the vehicle's measured power consumption during deployment.

Aspects of mission planning for autonomous vehicles can be found in the field of Operations Research (OR), where logistical planning problems are defined as optimisation problems and then solved. The Team Orienteering Problem (TOP) (Tsiligirides, 1984, Chao et al., 1996) is a good candidate for the modelling of standard AMV deployments where vehicles must visit operator-specified positions of interest in order to perform tasks (such as sampling the environment and performing intervention actions). Variants of the TOP have also been implemented for the planning of multi-AMV correlated scalar field sampling missions (Tsiogkas and Lane, 2018). Adapting the TOP formulation for deployment in uncertain environments, where the en-

ergy costs for vehicles to perform tasks is not deterministic, requires the TOP to be configured for Stochastic Weights (TOP-SW).

Evers et al. (2014) and Shang et al. (2016) proposed two-stage solutions to the Orienteering Problem with Stochastic Weights (OP-SW). The first stage selects a route for a singular vehicle based on the expected weight costs for each transition. In the second stage these weights are realised and a 'return home' recourse action is implemented if the realised total cost exceeds the total limit. The profit shortage cost (i.e. the number of points not visited because of the recourse action) in summation with the first stage's profit is used as a global objective function. Maximising the global objective function creates a route that maximises points collected and minimises the expected profit-shortage consequence. Evers et al. (2014) uses a OP-SW heuristic adapted from Sample Average Approximation (SAA) to evaluate the two-stage solution. SAA performs Monte Carlo simulation on the weights (which are random variables) to construct the objective function as a deterministic mixed-integer programming problem. While these two-stage solvers provide robust solutions, they are limited in their scope based on what actions the vehicle can take at any given moment. For example, the vehicle could choose to skip the current task if it is taking longer than expected to complete.

This paper continues to develop the energy-aware planner from Thompson and Galeazzi (2018) by implementing it onboard a real marine vehicle platform. Marine robots operate in a dynamic and uncertain environment that imparts non-linear and uncertain forces onto the vehicles. In section 4.2, we propose an AMV mission planner that is inspired by the two-stage method used in OR for solving the OP-SW, but adapted for *in situ* decision-making.

The first stage (section 4.2.2) computes the expected task sequence using the Monte Carlo sampling method in Thompson and Galeazzi (2018) *a priori* to vehicle deployment. The second stage (section 4.2.2) occurs during deployment of the AMV, and is computed locally onboard the vehicle. During the mission execution, the weights for each section of the plan are revealed sequentially. This, coupled with the potential for vehicle-to-shore communication dropouts, makes the two-stage solvers difficult to implement as SAA or other solvers are too computationally expensive to execute onboard the computer of an out-of-contact AMV. Instead, we propose a supervisor agent acting onboard the AMV that decides whether to enact one of several recourse actions arranged in a subsumption architecture (Brooks, 1986) style:

1. Continue current plan.
2. Skip the current task.

3. Request a replan from the shore mission planning agent.
4. Return to the rendezvous (home) position.
5. Emergency power saving mode.

To enable the supervisor to decide on one of these actions, three probabilistic metrics are proposed (section 4.2.3):

1. Confidence that the energy allocated for the current task has not been exceeded.
2. Confidence that the energy allocated for the current plan has not been exceeded.
3. Confidence that the energy capacity of the battery (or some fraction of it) has not been exceeded.

The confidence metrics are the result of computing the survival function of the predicted energy consumption distributions generated by the first stage planner, and using the measured energy consumption of the battery as input. In this context, the survival function provides an estimate of how likely an energy consumption measurement reading has exceeded a predicted task, plan, or battery distribution. An operator can then specify acceptable confidence thresholds for the supervisor that control the minimum confidence of the supervisor before a recourse action is activated.

A prototype ASV platform was designed with the specific purpose of testing the outlined two-stage planning approach (section 4.3). The ASV was deployed in a lake environment, where fluctuating winds produced uncertain external forces that were not directly available for consideration by the mission planner. During trials (section 4.5.1), combinations of confidence metrics were used to produce trajectories that conserved the original plan before returning home, and others that actively changed the plan to find achievable tasks.

To allow the supervisor agent to look ahead in time so that it can make recourse action decisions sooner, this paper also proposes a data-driven approach to forecasting the vehicle's energy consumption. Forecasting energy consumption has been achieved for ground robots through linear regression and Bayesian estimation (Sadrpour et al., 2013), and through encoding the mission tasks into a Long Short-Term Memory (LSTM) network (Hamza and Ayanian, 2017). Marine vehicle dynamics are non-linear, and the marine environment is much more dynamic and uncertain than terrestrial environments. In this respect, the use of non-linear regression models and probabilistic models are more likely to succeed in forecasting. LSTMs are an adaptation of Recurrent Neural Networkss (RNNs) that include input, output and forget gates

in order to overcome the vanishing gradient problem experienced by RNNs. LSTMs have seen significant success in sequential data problems such as handwriting recognition (Greff et al., 2017), weather forecasting (Zaytar and El, 2016), and ocean surface temperature forecasting (Caley et al., 2017) as they are able to identify and remember important features that influence the data later on.

In section 4.4, we propose a hybrid LSTM network control model to predict the motion of the vehicle, output of the vehicle's thrusters, and subsequent energy consumption. The LSTM networks were trained on the data gathered from the lake trials, and analysis of the hybrid energy forecaster shows that it is capable of reliably forecasting the energy consumption of the vehicle up to 10 seconds into the future.

4.2 Stochastic Programming Formulation

4.2.1 Original Mission Planner Definition

In Thompson and Galeazzi (2018), the multi-AMV mission planning problem was modelled as the TOP (Chao et al., 1996). The following definitions of a vehicle, task, and open and closed mission plans are presented for completeness:

$$V = (e_b, I_v) \tag{4.1}$$

$$T = (g, s, I_t) \tag{4.2}$$

$$M_o = (\mathcal{T}, \mathcal{V}, \mathcal{O}, P, Q, E) \tag{4.3}$$

$$M_c = (\mathcal{T}, \mathcal{V}, R, S, F) \tag{4.4}$$

where V is a vehicle, represented by a tuple containing the energy capacity of the battery (e_b) in Watt-hours (Wh) or Joules (J) and I_v is a tuple containing additional information about the vehicle (speed, operating domain, capabilities, etc.). T is a task, represented by a tuple containing the positional information of the task (g), the operator specified reward for completing the task (s), and I_t is a tuple containing additional information about the task (e.g. payload requirements, requisite and dependent tasks). To accommodate for missions where there are N_V vehicles and N_T tasks, $\mathcal{V}$ and $\mathcal{T}$ are defined as the accumulated set of defined V and T.

The first step of the planner is to use the above information to create the open mission, M_o,

which represents the complete domain that the planner searches through to obtain the closed mission M_c. P is a reference vector containing sequential integers that reference elements of $\mathcal{T}$. Q is a similar reference vector for $\mathcal{V}$. E is a zero-diagonal matrix of energy costs for transitioning between the ith and jth tasks ($\mathcal{T}_{P_{ij}}$).

The energy cost for E_{ij} is the result of performing a Monte Carlo simulation of size N on the marine vehicle dynamic model. Monte Carlo simulation of the model was necessary to capture the uncertainty of the hydrodynamic coefficients used in the model. The simulation first produces N time-varying sets of body forces required for the vehicle model to move along a reference trajectory. Each set of body forces are then decomposed into N sets of actuator allocations using a control allocation algorithm. Each set of actuator allocations are then converted to power consumptions through identified thrust-power relationships for each actuator. The summation of these actuator power consumption sets as well as the vehicle's hotel load produces N time-varying total power consumptions for the simulated vehicle models along the reference trajectory, $\mathcal{P}_k(t)$. The energy cost is the expected value (denoted by the operator $E[\cdot]$, not to be confused with the energy cost E) of the integral of these distributions with time:

$$E_{ij} = E\left[\frac{\sum_{k=1}^{N}(\int_{t_i}^{t_j}\mathcal{P}_k(t)\,\mathrm{d}t)}{N}\right] \tag{4.5}$$

$\mathcal{O}$ is a set of tuples that contain obstacle information necessary for collision avoidance path-planning and will not be considered further in this paper as it is tangential to the main question of energy planning. R is a N_V long set, each element of which contains a subset of P that represents the ordered sequence of tasks allocated to a vehicle. S is the set of rewards accumulated from completed tasks in R. F is a N_V long set, each element of which contains a subset of E that correspond to the energy costs for each task scheduled and allocated according to R.

The planner formulates the search for an optimum M_c into the following optimisation problem:

$$\begin{aligned} &\underset{M_c}{\text{maximise}} && \sum_{x_i \in S} x_i \\ &\text{subject to} && \sum_{y_i \in F_Q} y_i \leq e_b \in \mathcal{V}_Q \end{aligned} \tag{4.6}$$

where the goal is to maximise the reward collected in S while ensuring that the sum of energy costs in F do not exceed battery constraints of each corresponding vehicle (e_b).

4.2.2 Adaptation for Stochastic Weights

Even though the energy for each transition and task was obtained as a random variable through Monte Carlo simulation, the planner only uses the expected values (equation 4.5) and does not consider the full distribution of possible energy consumptions. This seems like a sensible choice as the expected value is the most likely amount of energy to be consumed for a given task transition (provided the distribution is Gaussian). However, this does not remove the chance that the energy consumption is more than expected, which could jeopardise the feasibility of the entire plan. In reality, these transition weights are complex and non-trivial to determine for certain, and depend upon the following:

1. Satisfactory identification of the vehicle's dynamic model.
2. Satisfactory identification of the vehicle's propulsion thrust/power relationship.
3. An accurate model of the vehicle's mechanical and electric efficiency up to the power source.
4. An accurate model of the wind, wave, and current forces acting upon the vehicle.
5. Well designed controllers that are able to track plan-generated reference trajectories.

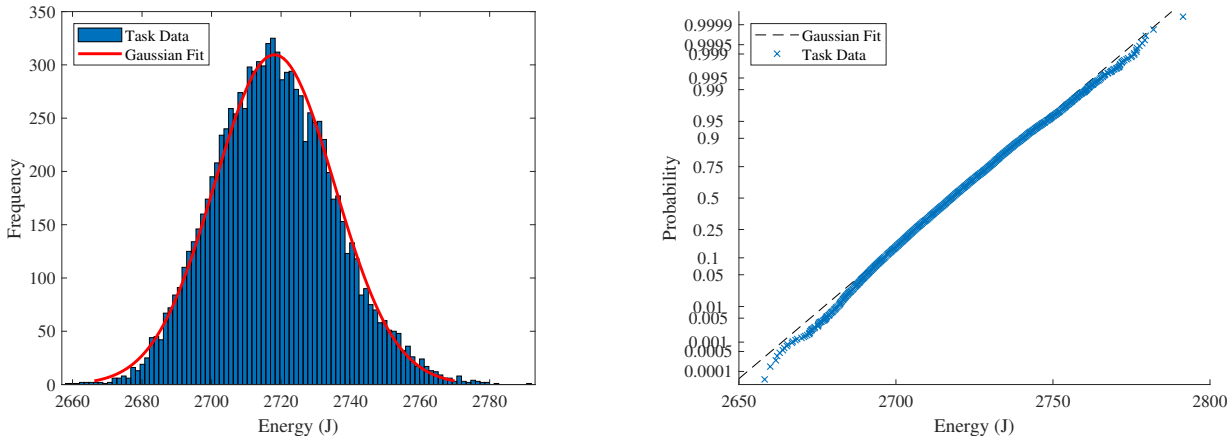
In particular, variance and unknown parameters within the environment model contribute to unpredictable behaviour in the controllers, leading to a higher variance in the *a priori* mission energy consumption prediction. Therefore, the planner must in some way accommodate for situations in which the realised energy consumption for a given task transition is greater than expected. The same can be said for the reverse situation where the realised consumption is less than expected.

To account for this uncertainty, OR researchers consider solutions to the OP-SW. A successful strategy for solving the OP-SW is to first solve the OP with the expected values of the weights. Then, once the vehicle is deployed on the initial route, a second stage solver keeps track of each transition's true weight once it has been realised. It then initiates a 'go to finish' or 'return home' recourse when the remaining transition costs plus the realised costs exceed the total OP cost allowance.

The mission planner performs Monte Carlo simulation upon a sampled vehicle model to obtain the task energy requirement distributions before the vehicle is deployed (the expected values of these distributions are used to form E in the original M_o definition). For the purposes of minimising computational resources in the solving of the OP-SW and also in minimising

communication overhead between the vehicle and shore, it would be advantageous to be able to parameterise the output distributions with a fitted standard distribution. The simplest fit to approximate the distribution (at least in number of parameters) is the Gaussian distribution, requiring just the mean and the variance.

Testing the output distributions of the Monte Carlo simulation for normality using the Andersen-Darling test statistic (at 5% significance) showed that the distributions are not from a Gaussian distribution. This means that the output distributions are not strictly Gaussian, and errors will have to be accepted if a Gaussian approximation is used. Consider the example distribution of the energy prediction for a transition task in figure 4.1. It is clear from 4.1b that the distribution loses correlation with the Gaussian fit at the upper and lower 5% boundaries ($x > 0.95$ and $x < 0.05$). On close inspection, the Gaussian fit overestimates the likelihood of the task's energy requirement towards the lower 5% boundary, and underestimates the likelihood of the requirement towards the upper 5% boundary. This means that if a Gaussian fit is used, the planner will have a tendency to use an optimistic prediction of the energy consumption due to the approximation error. Given these limitations, it must be acknowledged that approximating the distribution as Gaussian is an engineering trade-off between accuracy of the model prediction and the practical limitations of computation and communication in the field.



(a) Task energy histogram with Gaussian fit.

(b) Task energy probability plot with Gaussian fit.

Figure 4.1: Plots of an example task energy distribution generated from 10,000 simulations of varying vehicle dynamics models using equation 4.5. Figure 4.1b shows a correlation with a Gaussian distribution between the upper and lower 5% boundaries.

In this paper, we accept the implications of using a Gaussian approximation, and model the generated distributions with only their mean and standard deviation. E is then redefined as two separate matrices, μ_E and σ_E, representing the mean and standard deviation of the ijth task transition respectively. M_c is then solved by the planner using μ_e instead of E. F

is similarly redefined into μ_F and σ_F, which gives the means and standard deviations of the ordered sequence of transition weights for each vehicle respectively.

The stochastic energy prediction, H, is defined as a random variable pertaining to the task, plan or battery capacity as follows:

$$H \sim \mathcal{N}(\mu, \sigma^2) \tag{4.7}$$

$$H_t \sim \mathcal{N}(\mu_{F_i}, {\sigma_{F_i}}^2) \tag{4.8}$$

$$H_p \sim \mathcal{N}(\sum \mu_F, \sum {\sigma_F}^2) \tag{4.9}$$

$$H_b \sim \mathcal{N}(\mu_{e_b}, {\sigma_{e_b}}^2) \tag{4.10}$$

where H_t is the energy prediction for a particular task, H_p is the energy prediction for the summation of tasks to be performed, and H_b is a random variable obtained based on battery discharge/recharge data.

Naive Energy Consumption Certainty Estimation

The vehicle is equipped with sensors to measure the voltage and current consumption close to the battery terminals. The measured energy consumption of the vehicle is calculated by performing numerical integration using the trapezoidal rule of the measured power consumption over the time interval $\Delta t = t(k) - t(k-1)$:

$$E_m(k) = \frac{P_m(k) + P_m(k-1)}{2}\Delta t + E_m(k-1) \tag{4.11}$$

With the energy consumption prediction now represented as a Gaussian random variable (H), a simple metric to determine the likelihood that the vehicle has consumed more than the prediction is the survival function:

$$S_H(E_m(k)) = P(H > E_m(k)) = 1 - \int_{-\infty}^{E_m(k)} \frac{1}{\sqrt{2\pi\sigma_H^2}} e^{-\frac{(x-\mu_H)^2}{2\sigma_H^2}}\, \mathrm{d}x \tag{4.12}$$

The survival function metric allows the operator to specify a lower limit (δ) for the supervisor based on the likelihood that $E_m(k)$ is not greater than H. If $S_H < \delta$, then the supervisor will activate a recourse action behaviour. For $\delta > 0.5$, the operator is encouraging the supervisor to be conservative and activate recourse actions earlier and *vice versa* for $\delta < 0.5$.

4.2.3 Recourse Actions

By implementing S_H and the δ condition across different energy consumption scales, several levels of decision making for the supervisor can be designed based on the expected operations of a deployed vehicle. For example, when the vehicle switches to its own power supply, it must then keep track of the energy consumed when compared to the estimated energy capacity of the vehicle, e_b. When the supervisor commences a plan given to it by the mission planner, it must compare the energy consumed against the predicted total energy consumption of the plan. Finally, each plan is a sequence of tasks, each of which should be considered on the task energy prediction scale.

To formalise this we defined separate datum points for the battery, plan, and task energy scales.

1. On vehicle power source mode switch to battery (battery datum), o_b
2. On commencement of a plan (plan datum), o_p.
3. On commencement of a task (task datum), o_t.

As the vehicle progresses through a mission, task E_t, plan E_p, and battery E_b scales are simultaneously evaluated in the equations below.

$$E_b(k) = E_m(k) - o_b \tag{4.13}$$

$$E_p(k) = E_m(k) - o_p \tag{4.14}$$

$$E_t(k) = E_m(k) - o_t \tag{4.15}$$

These energy measurements are then compared with their respective H energy predictions (equations 4.8-4.10) based on the survival function activation criteria (equation 4.12). As an additional fail-safe, we also place a hard limit on the minimum measured voltage of the battery, V_{lim}. The recourse actions and their activation conditions are listed below:

1. $S_{H_t}(E_t(k)) < \delta_t$: skip task heuristic.
2. $S_{H_p}(E_p(k)) < \delta_p$: replan heuristic.
3. $S_{H_b}(E_b(k)) < \delta_b$: return to rendezvous (home) position.
4. $V_m(k) < V_{lim}$: emergency power saving mode.
5. Otherwise: continue current plan.

The first and second recourse action activations are described in the following subsections. The third activation commands the vehicle to travel to the home point. The fourth activation is an emergency fail-safe mode triggered when the voltage of the battery has dropped below the minimum voltage requirement of the thrusters. The vehicle shuts down the motors and is stranded.

Task Skip Heuristic

The task skip heuristic enables when the task survival function is below the task survival threshold. The supervisor performs a naive linear estimate of the energy remaining for the current task, E'_T by:

$$E'_t = \frac{\mathcal{S}_{bc}}{\mathcal{S}_{ac}} E_t \tag{4.16}$$

where $\mathcal{S}_{bc}$ is the distance remaining on the predicted trajectory from the vehicle's current position to the goal, and $\mathcal{S}_{ac}$ is the total distance of the predicted trajectory. A decision whether to skip or not skip the remainder of the current task is given in the following algorithm:

Algorithm 3: Task skip heuristic.

input : Energy remaining for current task E'_t
Reward of current task s_t
Set of rewards of tasks remaining S
Set of energy costs of remaining tasks F

output: True: skip task or False: keep task

```
1  e = 0;
2  s = 0;
3  [F', I] =sort(F);
4  S' = S(I);
5  for i ← 1 to dim_F(F') do
6  |   e ← e + F'(i);
7  |   s ← s + S'(i);
8  |   if e < E'_t then
9  |   |   if s > s_t then
10 |   |   |   return True;
11 |   else
12 |   |   return False
```

The task skip heuristic sorts in ascending order the remaining tasks for the current plan by

predicted energy cost (F_{ij}), and iteratively aggregates the energy cost of this sorted list until it exceeds E'_t. If the accumulated reward of the sorted, remaining tasks exceeds the reward for the current before this condition then the algorithm returns true (i.e. skip the task). Otherwise the algorithm returns false (i.e. keep the task).

Replan Action

During deployment, the vehicle keeps a track of the tasks that were completed and the tasks that were skipped due to the task skip recourse action. Upon activation of the replan recourse action, the vehicle first must determine if it can communicate with the first stage of the mission planner. If communication is successful, it sends the request $\mathcal{R}$ to the mission planner, containing the following information:

$$\mathcal{R} = (\mathcal{C}, \mathcal{D}, \mathcal{E}, L, E(k)) \tag{4.17}$$

where $\mathcal{C} \subset R$ is the set of completed tasks, $\mathcal{D} \subset R$ is the set of skipped tasks, $\mathcal{E}$ is the set of final energy measurements for each completed task (E_t) and L is the location of the vehicle at the time of replan request. Some of the elements in $\mathcal{E}$ will contain the consumed energy from tasks that were skipped previous to it. The vehicle then holds its current position while the mission planner generates a new solution from $\mathcal{R}$ and updates M_c. Once the vehicle receives the updated M_c, it begins the new plan. If the vehicle is not within communication range, it activates the 'return home' recourse action. Ideally, the replanning recourse would happen entirely onboard the vehicle. However, due to the computational constraints of current small form factor embedded computers that typically run the software of AMVs, the replanning steps must be outsourced to an external computer (such as a shoreside system) that can handle the planning requirements.

One potential method for enabling online replanning on a low-cost embedded system would be to create a lightweight planning agent that only uses the task information given in the original plan to generate a replan solution. This means that not all potential tasks will be considered, but the vehicle would then be able to create a new plan based on a subset of the old. This also ensures, in a multi-AMV deployment, that each vehicle would be guaranteed not to create conflicts with other vehicles by allocating itself an already allocated task. This comes with the caveat of restricting each vehicle's knowledge of the global mission state, meaning that vehicles will be unable to act on tasks that weren't initially given to them. This increases the risk of mission failure due to local vehicle failures. A distributed planning architecture, such as described in Zlot (2006) and Sotzing et al. (2007), would enable vehicles to actively give, take

and swap tasks according to their replan actions.

Upon receiving $\mathcal{R}$ from the vehicle, the first stage planner formulates a new M_o based on the tasks in the old M_c that were neither skipped nor completed. This reduces the size of E that has to be searched through, because the rows and columns of the previous E that reference starting from, or moving to completed or skipped tasks can be deleted. The energy constraint (e_b from equation 4.6) is replaced with the previous plan's energy prediction minus the energy consumed during deployment (replace e_b with $\mu_{H_p} - E(k)$ in equation 4.6). Additionally, the planner must redefine E_{1j} and E_{i1} with an energy distribution prediction based on the provided L, which is the new starting point of the vehicle in the new plan. By reducing the size of E and only performing Monte Carlo simulation on the subset of trajectories that start at L, the replanning process time is a fraction of the initial plan generation time.

4.3 System Description

The full system is comprised of two components: the shoreside mission planner and HMI, and the ASV[2]. The shoreside systems and ASV communicate with each other over a Wi-Fi network, with a Radio Controller (RC) included as a manual override backup. Both systems were built using the Robot Operating System (ROS) middleware.

4.3.1 Shoreside Subsystem

Figure 4.2 depicts the major components of the shoreside system. The first stage of the mission planner described in sections 4.2.1-4.2.2 was developed in MATLAB and extended with the robotics system toolbox to allow it to interface with the ROS network. The operator provides the mission planner with $\mathcal{T}$ and $\mathcal{V}$, and the mission planner computes an optimal plan that it sends to the ASV via a ROS message. During mission run-time, the planner listens for planning requests from the ROS communication layer.

[2]The ASV system framework is open-source and has been made available at https://github.com/FletcherFT/asv_framework

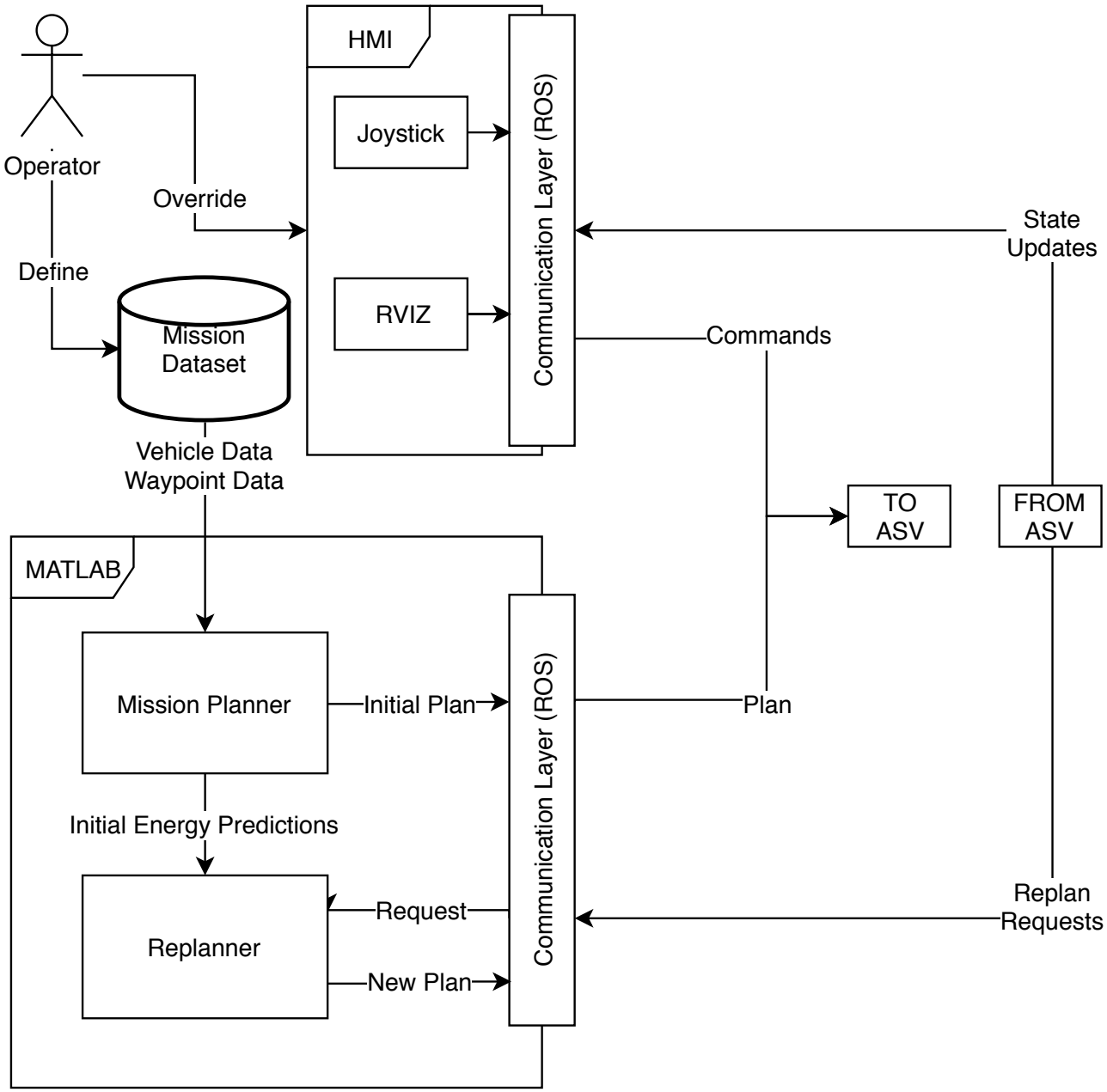


Figure 4.2: Layout of the shoreside system.

The HMI (see figure 4.3) is used by the operator to sanity check the navigation of the ASV by seeing on a map the vehicle's own estimate of its position and orientation. The HMI was built in ROS using RViz and a satellite map plugin. Override waypoint commands can be sent to the vehicle by clicking and dragging a position and orientation on the map, causing the vehicle to pause the current mission and navigate to the specified position. The operator can also see the vehicle's progress on the current mission red and green markers that indicate if a waypoint has been visited or not. A simple Graphical User Interface (GUI) to send start/pause mission commands, calibration commands to the navigation system, and Proportional Integral Derivative (PID) tuning commands to the control system was also implemented.

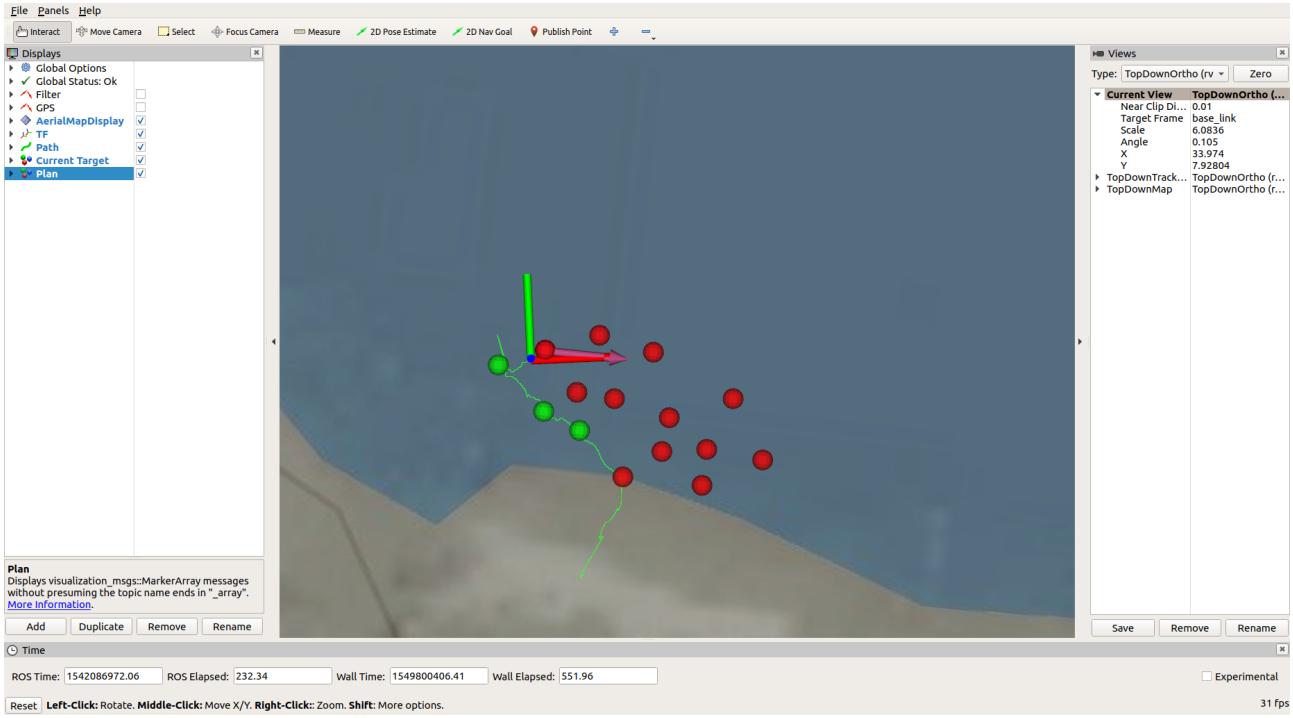


Figure 4.3: Screenshot from the HMI component of the shoreside system. The vehicle's position and orientation is provided by the red/green/blue axis, incomplete waypoints are shown as red spherical markers, completed tasks are shown in green. The selected waypoint for completion is shown with a red marker and fuchsia arrow.

4.3.2 ASV Subsystem

A simple model-scale box-barge hull was converted into a prototype autonomous platform for the purposes of testing planning, guidance, navigation and control algorithms. The vehicle, pictured in figure 4.4, has three actuators: one tunnel thruster located towards the bow and two azimuth thrusters located at the transom. This configuration means the vehicle can move independently in forwards (surge), sideways (sway), and heading (yaw) motions. To take advantage of this, an autopilot and a dynamic positioning controller were designed to control the vehicle in two different operating modes: transition mode (autopilot), and hold position mode (dynamic positioning). The ASV is equipped with a u-blox LEA-6H GPS for position feedback, and a Redshift UM7 IMU for orientation, angular velocity, and acceleration feedback. Velocity measurements from the GPS, such as course-over-ground information, were not included for state estimation as the LEA-6H obtains velocity information from the derivative of the position (not through an independent measurement such as Time-Differenced Carrier Phase).

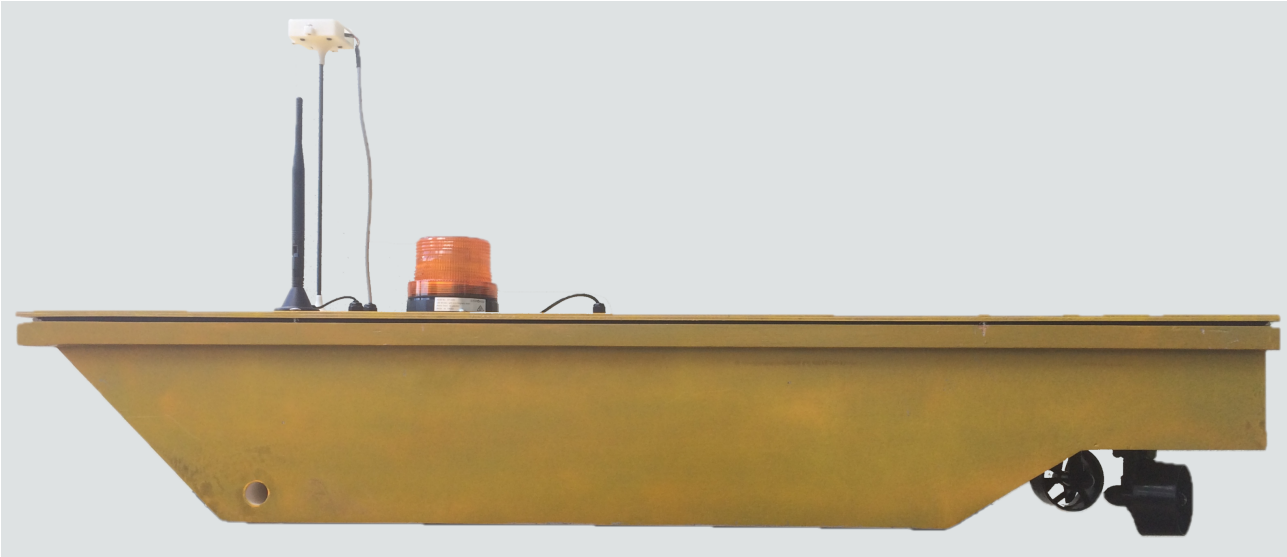


Figure 4.4: Profile view of the ASV. Two fully rotatable 70 mm thrusters are positioned on the aft transom. A tunnel thruster is positioned near the bow. The GPS and Wi-Fi communication masts are positioned on the deck outside of the water resistant enclosure. During trials, the IMU was also moved onto the deck to reduce magnetic interference from the DC motors.

Figure 4.5 depicts the general subsystems of the ASV. At the top of the process hierarchy is the supervisor agent, which has several roles. Firstly it is designed to receive and execute missions plans from the shoreside mission planner. During mission execution it configures the ASV for the current task by passing waypoint information to the guidance system and sending configuration commands to the control system. It uses the energy consumption information provided by the energy monitor to perform the recourse action decision-making described in sections 4.2.2-4.2.3. Lastly, it accepts operator commands and overrides and subsumes all other activities.

The lower levels of the ASV system are centered around the Guidance, Navigation and Control paradigm described in Fossen (2011a). The navigation module consists of drivers for reading the GPS and IMU, and an Extended Kalman Filter (EKF) state estimator. The EKF is based on a 2D constrained general point kinematic model described in Moore and Stouch (2014). It fuses the GPS and IMU measurements into an estimate of the vehicle's position (in UTM coordinates), orientation (following an East-North-Up convention), and angular and linear velocities in the body-frame of the vehicle (following a Forward-Port-Up convention). The chosen framing conventions differ from the marine robotics standard of using North-East-Down because the core ROS coordinate system operates in East-North-Up.

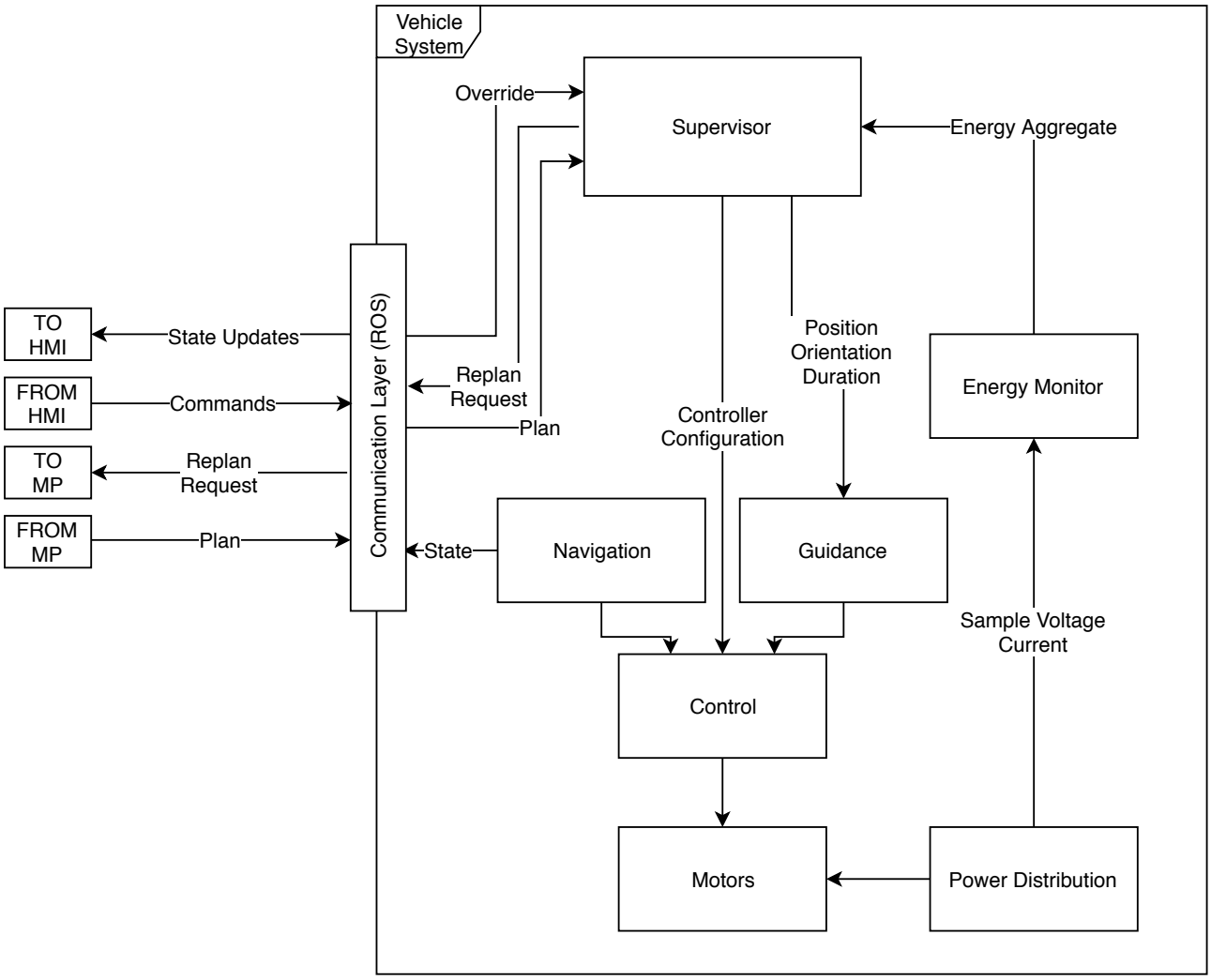


Figure 4.5: Subsystem layout of the ASV.

The guidance module uses the Line of Sight (LOS) guidance controller from Fossen (2011b) with a custom piecewise function to provide speed and heading errors for the control module to regulate to zero. The guidance distance and heading error, D and ψ_e are:

$$D = \sqrt{(x_d - x)^2 + (y_d - y)^2} \tag{4.18}$$

$$\psi_e = \text{atan2}(y_d - y, x_d - x) - \psi \tag{4.19}$$

where the terms are depicted in figure 4.6.

The autopilot component within the control module requires a forward speed error, which is

calculated in the guidance module as follows:

$$u_d(D, r_a, \psi_e) = \begin{cases} 0 & D \leq r_a \\ 0 & |\psi_e| > \frac{\pi}{5} \\ U & D > r_a, |\psi_e| \leq \frac{\pi}{5} \end{cases} \tag{4.20}$$

$$u_e = u_d - u \tag{4.21}$$

where r_a is the radius of acceptance (i.e. the maximum distance the vehicle can be from the waypoint). U is a forward speed setpoint that the operator specifies in I_v. Equation 4.20 contains conditional setpoint changes to slow the vehicle down and reduce its turning circle for large guidance heading errors (arbitrarily defined as anything larger than $\frac{\pi}{5}$ radians), and to stop the vehicle entirely when it has reached the radius of acceptance for the waypoint. When the waypoint is reached, the guidance module requests a new waypoint from the supervisor or idles if there is none.

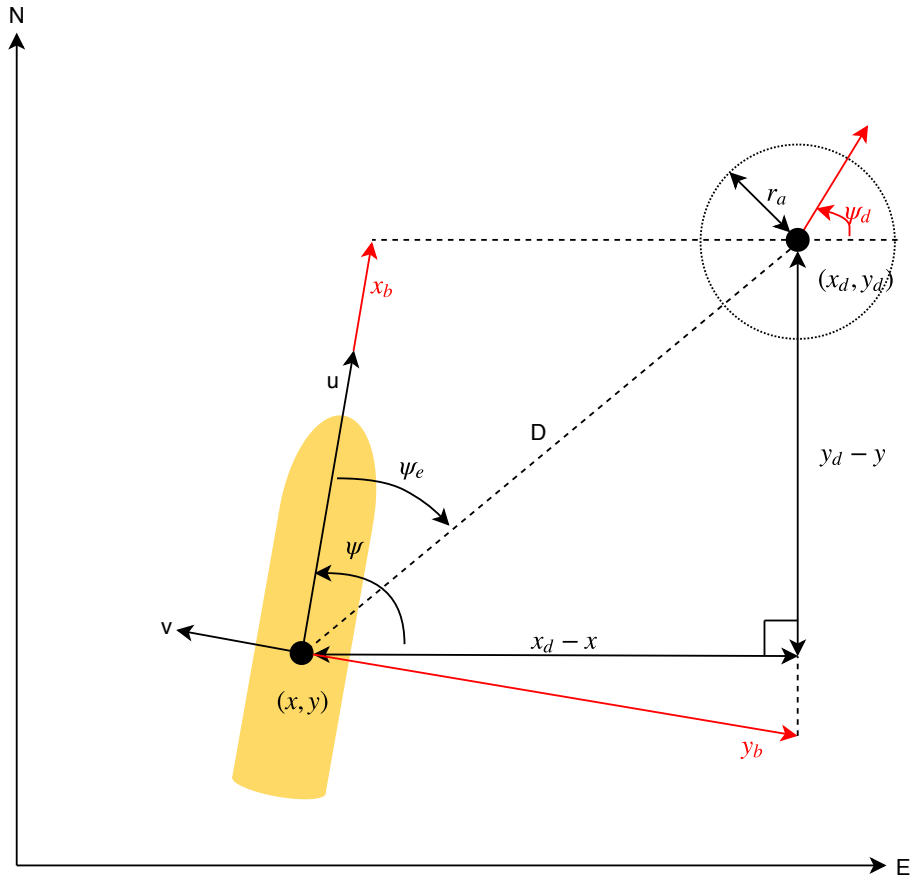


Figure 4.6: Geometry that the LOS guidance controller uses to calculate the autopilot and dynamic positioning contoller errors. The autopilot error signals depend on ψ_e and D. The dynamic positioning error signals depend on x_b, y_b, and ψ_d. The red lines are the projection of the ASV body-frame coordinates onto the position errors, which are used to calculate x_b and y_b.

When the ASV is in dynamic positioning mode, the guidance module provides the control module with longitudinal and transverse position errors by rotating the inertia-frame components of D into the body-frame:

$$R = \begin{bmatrix} -\sin\psi & 0 & 0 \\ 0 & \sin\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.22}$$

$$\begin{bmatrix} y_b \\ x_b \\ \psi_b \end{bmatrix} = R \begin{bmatrix} x_d - x \\ y_d - y \\ \psi_d - \psi \end{bmatrix} \tag{4.23}$$

where x_b and y_b are the body-frame position errors as depicted in figure 4.6. ψ_b is the error between the desired vehicle heading and the current vehicle heading.

The control module consists of an autopilot controller, a dynamic positioning controller, and a control allocation algorithm. The autopilot and dynamic positioning controllers regulate the speed and heading, and position and heading respectively by outputting a vector of commanded body forces in the surge and sway directions, and a yawing moment ($\boldsymbol{\tau^c}$). The autopilot controller is a multi-input, multi-output PID controller that outputs $\boldsymbol{\tau^c}$ as:

$$\tau_X^c = K_p u_e + K_i \int u_e \, \mathrm{d}t + K_d \frac{\mathrm{d}u_e}{\mathrm{d}t} \tag{4.24}$$

$$\tau_Y^c = 0 \tag{4.25}$$

$$\tau_{N_Z}^c = K_p \psi_e + K_i \int \psi_e \, \mathrm{d}t + K_d \frac{\mathrm{d}\psi_e}{\mathrm{d}t} \tag{4.26}$$

The transverse velocity is not regulated, which can cause the vehicle to drift sideways when turning.

The dynamic positioning controller calculates $\boldsymbol{\tau^c}$ through the following:

$$\tau_X^c = K_p x_b + K_i \int x_b \, \mathrm{d}t + K_d \frac{\mathrm{d}x_b}{\mathrm{d}t} \tag{4.27}$$

$$\tau_Y^c = K_p y_b + K_i \int y_b \, \mathrm{d}t + K_d \frac{\mathrm{d}y_b}{\mathrm{d}t} \tag{4.28}$$

$$\tau_{N_Z}^c = K_p \psi_b + K_i \int \psi_b \, \mathrm{d}t + K_d \frac{\mathrm{d}\psi_b}{\mathrm{d}t} \tag{4.29}$$

where ψ_d is the desired heading for the vehicle.

The control allocation algorithm is a quadratic programming solver formulated as the constrained fixed-angle thruster allocation problem described in Fossen (2011c). Although the thrusters are fully rotatable, power consumption can be reduced by ensuring that the azimuth thrusters only rotate when the supervisor switches the control mode from autopilot to dynamic positioning (or *vice versa*). Two thruster configurations were defined for each control mode. When in autopilot mode, the thrusters are oriented parallel to the longitudinal axis of the ASV. When in dynamic positioning mode, the thrusters are swivelled $\pm 45^\circ$ to either side of the longitudinal axis. The control allocation solution yields both the commanded thrusts for each thruster, and the achieved body forces and yawing moment provided by the solution ($\boldsymbol{\tau}^{\boldsymbol{a}}$). The commanded thrusts are converted to a vector of electronic speed controller duty cycles ($\boldsymbol{d}$), which are outputted to the motor drivers for each respective thruster.

4.4 Energy Forecasting

In section 4.2.2 a naive method was presented that calculated a survival metric based on the aggregated energy measurements. Through this method the operator safeguards against the future by providing a margin of safety on the survival thresholds, making the supervisor trigger recourse actions according to the likelihood that the vehicle has used the planned energy for a task, plan, or the battery. One issue with this is that the recourse decision is made at the time of the energy measurement and runs the risk of being activated too late. A forecasting approach could be used *in situ* to predict ahead of time if a survival threshold will be crossed, allowing the supervisor to initiate recourse actions earlier and save energy in the process.

Forecasting the power consumption of the vehicle is a three-stage forecasting process that requires:

1. Prediction of the vehicle's kinematics.
2. Prediction of the vehicle's control response for a given task and kinematic state.
3. Prediction of the vehicle's power consumption given the control response of the vehicle.

The following subsections detail a hybrid state prediction and control process, using data-driven machine learning models for the kinematic state and power consumption prediction, and classic PID control processes for the control response calculations. A simplified block diagram summarising the major stages of the process is provided in figure 4.7.

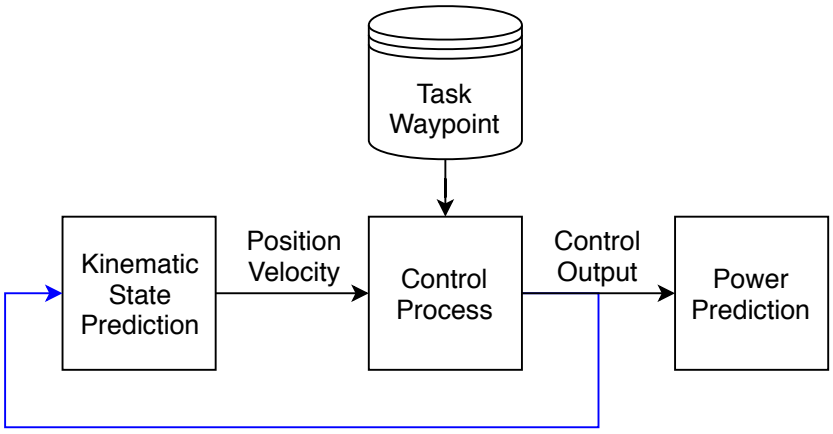


Figure 4.7: Simple view of the hybrid power prediction process. The first block predicts the new kinematic state of the vehicle given the control output as external feedback. The second block receives the new kinematic state and a waypoint reference from the task data and computes a control output. The third block receives the new control output and predicts the power consumption of the vehicle. The main feedback loop (i.e. setting $k = k + 1$) is coloured blue.

4.4.1 Kinematic State Prediction

The ASV is expected to operate in an environment without large rolling, pitching, or heaving motions, so the kinematics simplify to 3 degrees of freedom (two in translation, and one in rotation). The kinematic state of the vehicle is defined as:

$$\boldsymbol{\eta} = [x, y, \psi]^T \tag{4.30}$$

$$\boldsymbol{\nu} = [u, v, r]^T \tag{4.31}$$

where $\boldsymbol{\eta}$ is the vehicle's position in the inertia-frame. In this case the x and y coordinates are the vehicle's latitude and longitude projected into UTM, and ψ is the vehicle's heading relative to East following a East-North-Up convention. $\boldsymbol{\nu}$ is the ASV's body-frame linear velocities in the longitudinal (u) and transverse (v) directions and the angular turning rate in yaw (r). The vectorised kinetic model for marine vehicles is:

$$\boldsymbol{\tau}^a = \mathbf{M}\dot{\boldsymbol{\nu}} + (\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}))\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) - \boldsymbol{\tau}^e \tag{4.32}$$

where $\boldsymbol{\tau}^a$ is the vector of achieved body-fixed forces outputted by the controller in longitudinal and transverse directions, and the commanded yawing moment. $\mathbf{M}$ is a matrix containing the summation of the added mass and inertia terms of the vehicle. $\mathbf{C}(\boldsymbol{\nu})$ is an array containing the summation of the rigid body and added mass Coriolis and centripetal terms as a function of $\boldsymbol{\nu}$. $\mathbf{D}(\boldsymbol{\nu})$ is a matrix of hydrodynamic damping terms (i.e. the drag coefficients) as a function

of $\boldsymbol{\nu}$. $\mathbf{g}$ is the vector of hydrostatic forces according to the vehicle's position and orientation within the water column. $\boldsymbol{\tau}^{\boldsymbol{e}}$ is the sum total of body-frame referenced environmental loads acting on the vehicle.

The vehicle's kinematic states are predicted according to evaluation of the kinetic model in equation 4.32. This is a complex process as both the manoeuvring loads and environmental loads are non-linear and may not be able to be accurately observed *in situ*. The constant parameters in $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{D}$ can be identified through captive model testing of the vehicle, but must be updated if modifications to the vehicle (such as payloads) are introduced. Calculation of $\boldsymbol{\tau}^{\boldsymbol{e}}$ also requires sensing and observation of the environment (wind, current, and waves) which this ASV is not equipped to perform.

An alternative to model-based kinematic state prediction is to perform supervised learning on collected data to obtain an approximation of the equation 4.32. LSTM networks have seen success in learning predictions from time series data in other domains, making it a good candidate for predicting the trajectory and control time series data. The LSTM design in figure 4.8 performs the following operation:

$$\boldsymbol{\nu}_{k+1} = f(\boldsymbol{\nu}_k, \psi_k, \boldsymbol{\tau^a}_k) \tag{4.33}$$

where ψ_k and $\boldsymbol{\tau^a}_k$ are chosen as additional explanatory variables. Due to the internal feedback nature of the LSTM architecture, the network may also learn relationships between the change in and aggregate of $\boldsymbol{\nu}$ over time ($\dot{\boldsymbol{\nu}}$ and $\boldsymbol{\eta}$) but this is not guaranteed. $\dot{\boldsymbol{\nu}}$ is not included as the measured accelerations provided by the IMU are too noisy. The x and y components of $\boldsymbol{\eta}$ are also not included as they are not bounded by upper and lower limits (unlike ψ which is bound between $\pm\pi$ radians), which would make it harder for the LSTM to generalise.

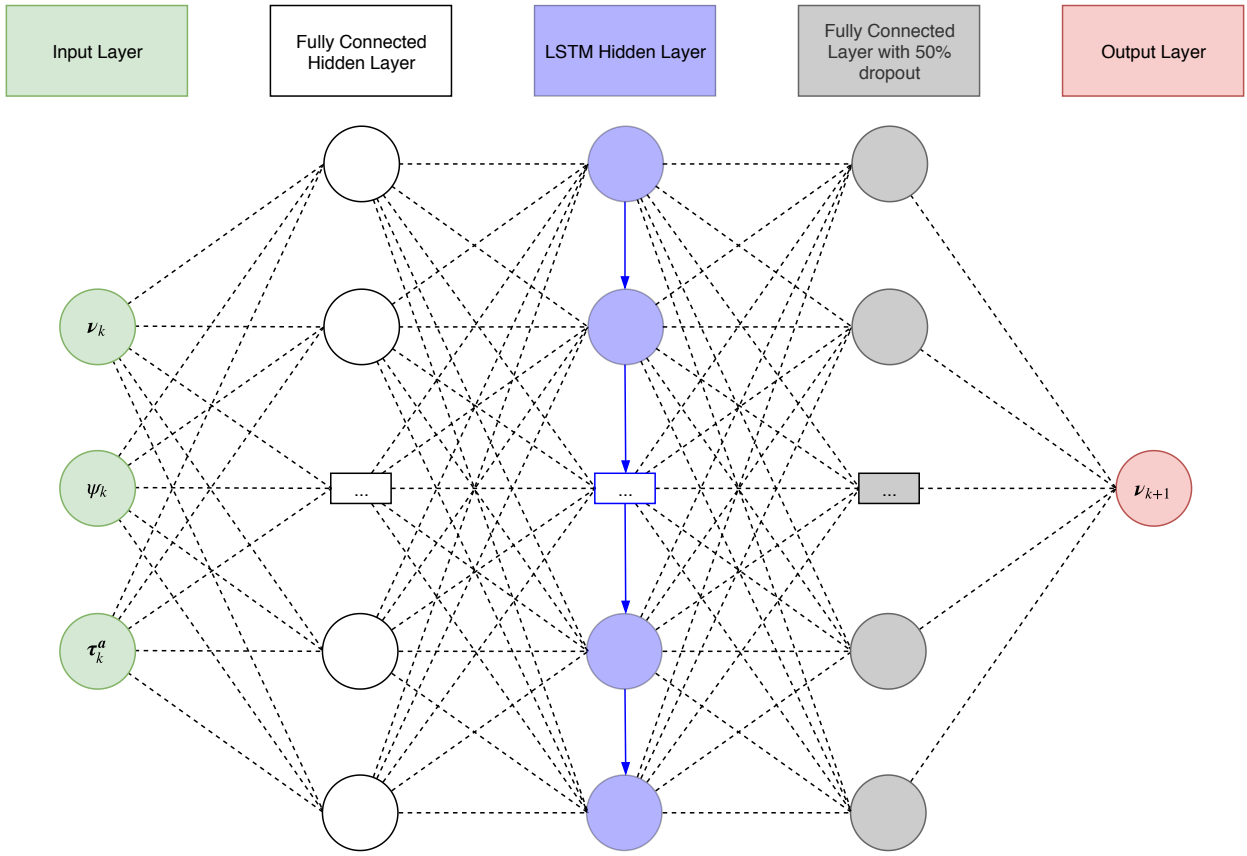


Figure 4.8: Network layout of the $\boldsymbol{\nu}$ state prediction LSTM. The network accepts the current body velocity vector $\boldsymbol{\nu}_k$, the current yaw heading ψ_k, and the achieved vector of body forces $\boldsymbol{\tau^a}_k$. The network consists of a latent space hidden layer, the LSTM layer, another fully connected layer with a 50% dropout to reduce the chance of overfitting, and then the output layer (which is $\boldsymbol{\nu}_{k+1}$).

4.4.2 Power Consumption Prediction

As discussed in section 4.2.1, the power consumption of the ASV depends upon the commanded thruster outputs, the hotel load, and the efficiency of the electrical distribution system. Identification of each of these contributors can provide a good estimate of the total power consumption of the vehicle. However, complex transient effects that occur on the distribution system during changes in thruster output, and the effect of hydrodynamic loading on the thrusters during changes in output are not typically modelled. Once again, a data-driven approach using an LSTM network (see figure 4.9) may be able to capture these effects.

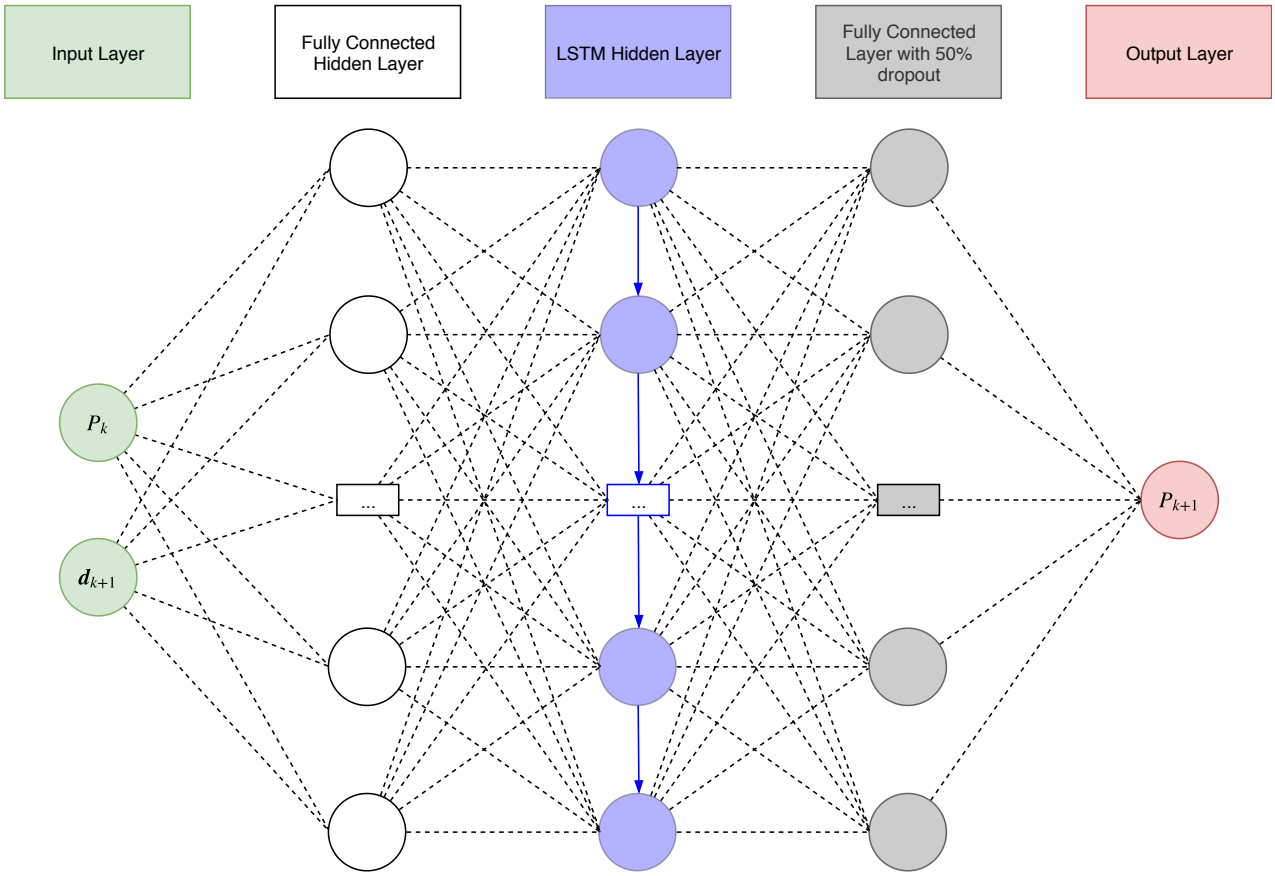


Figure 4.9: Network layout of the power prediction LSTM. The network accepts the current power P_k and the predicted commanded duty cycles of the thrusters from the control allocation block ($\boldsymbol{d}_{k+1}$). The network consists of a latent space hidden layer, the LSTM layer, another fully connected layer with a 50% dropout to reduce the chance of overfitting, and then the output layer (which is P_{k+1}).

The power prediction network performs the following operation:

$$P_{k+1} = f(P_k, \boldsymbol{d}_{k+1}) \tag{4.34}$$

where $\boldsymbol{d}_{k+1}$ is the vector of commanded duty cycles for all thrusters as outputted by the control process prediction. Duty cycle was selected over force because duty cycle is bounded between 0 and 1, making it easier for the LSTM to generalise.

4.4.3 Hybrid Energy Forecaster Model

The kinematic state of the vehicle is used by the control process outlined in section 4.3 to obtain the achieved body forces to regulate the kinematic state to specific set points that reach the task waypoint following the LOS guidance controller. Figure 4.10 shows the full process of the hybrid energy forecast model. The $\boldsymbol{\nu}$ prediction LSTM first predicts the body-frame velocities of the vehicle at $k+1$. Then $\boldsymbol{\nu}_{k+1}$ is integrated and transformed into the inertia frame to

obtain $\boldsymbol{\eta}_{k+1}$. The task waypoint, $\boldsymbol{\nu}_{k+1}$, and $\boldsymbol{\eta}_{k+1}$ are inputted into the LOS guidance controller to obtain the forward speed error, u^e_{k+1}, and the LOS heading error, ψ^e_{k+1}.

The autopilot controller functions as described in equations 4.24-4.26. The gains for the controllers are copied from the gains used in the ASV control system. The autopilot controller outputs the vector of desired body forces and moments ($\boldsymbol{\tau^c}_{k+1}$), which is then inputted into a copy of the ASV's control allocation process. The control allocation process outputs the achieved body forces and moments, $\boldsymbol{\tau^a}_{k+1}$, which is fed back into the $\boldsymbol{\nu}$ prediction LSTM. The duty cycles for the thrusters, $\boldsymbol{d}_{k_1}$, is also calculated by the control allocation process and is used by the power prediction LSTM to obtain P_{k+1}. $\boldsymbol{\tau^a}_{k+1}$ is fed back into the $\boldsymbol{\nu}$ prediction LSTM for the next prediction cycle. The energy forecast can then be computed by aggregating the forecast power from equation 4.34 substituted into equation 4.11.

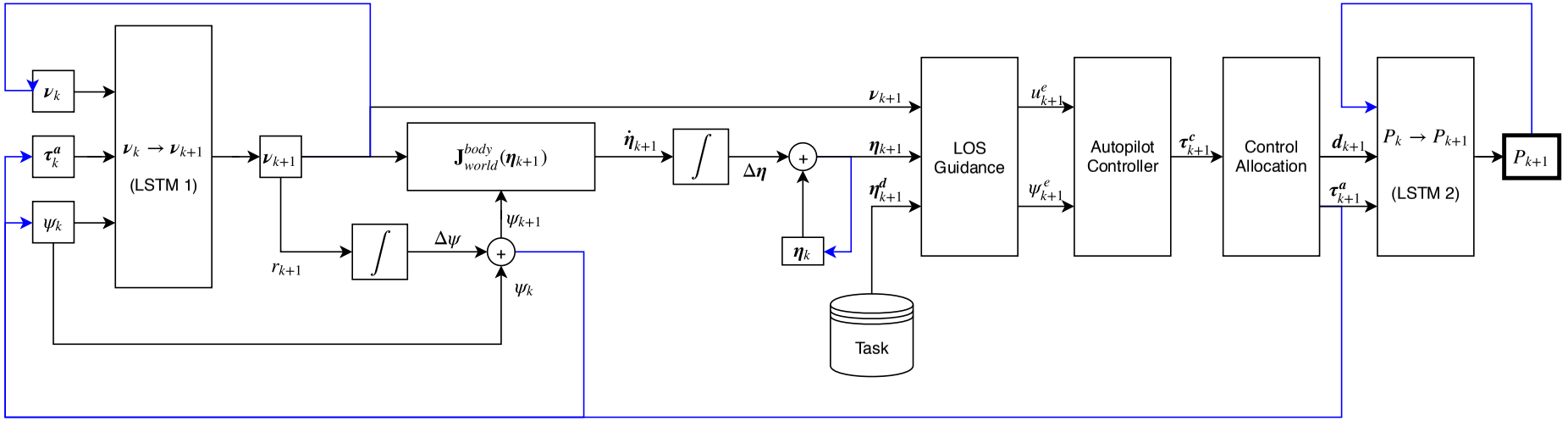


Figure 4.10: Full block diagram of the hybrid LSTM/control process power forecaster. The $\boldsymbol{\nu}$ state prediction LSTM produces $\boldsymbol{\nu}_{k+1}$ predictions which are then transformed to the inertia frame and integrated to obtain the vehicle's current position, a LOS guidance function calculates the setpoint for the forward speed controller and the relative bearing error, which feeds into the autopilot controller. The autopilot outputs a commanded body force and moment vector $\boldsymbol{\tau^c}_{k+1}$ to the control allocation algorithm, which determines the duty cycle for each thruster ($\boldsymbol{d}_{k+1}$). Finally the power prediction LSTM uses the power from the current step P_k and the predicted thruster duty cycle $\boldsymbol{d}_{k+1}$ to predict P_{k+1}. The blue lines indicate the feedback paths where k is set to $k+1$ for the next time step.

4.5 Results of Field Trials and Energy Forecasting

Figure 4.11 details the mission data sets that were generated within a Geographic Information Systems package. The close, medium, and long range missions were specified as collections of waypoints with no specific order ($\mathcal{T}$ from section 4.2.1). The rectangle mission has a human-defined order as it is a calibration test for the ASV's autopilot controller. Each mission was run several times with varying δ_p and δ_t thresholds, which are the primary criteria used by the supervisor for making *in situ* decisions. Weather data was obtained from Bureau of Meteorology (2018) during post-processing of the data. Section 4.5.1 presents two example runs where the behaviour of the vehicle is drastically different due to the manipulation of these thresholds.

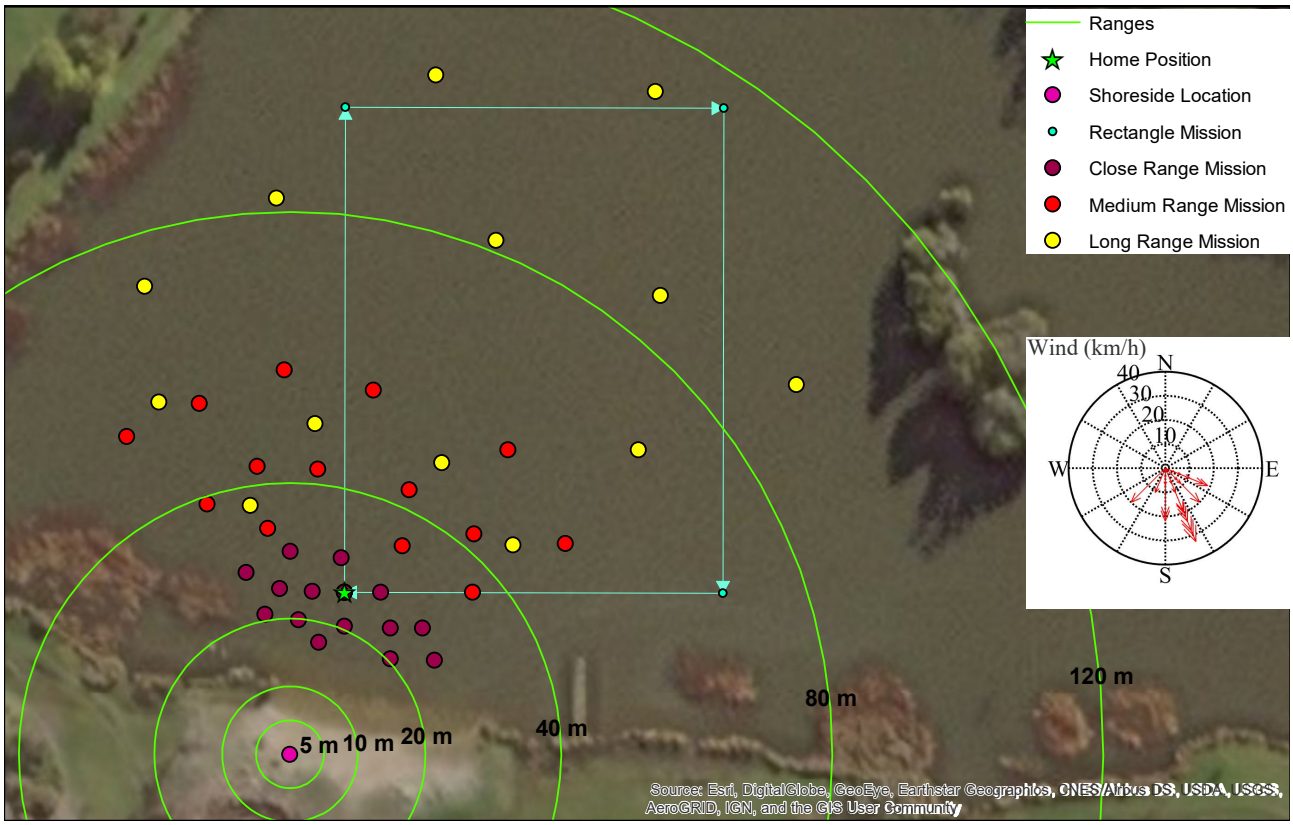


Figure 4.11: Summary of mission datasets generated for Lake Waverley. All missions except for the rectangle mission are not given a sequence by the operator. The mission planner creates a sequence during plan generation. Wind data was obtained from a local weather station 5 km from the lake, providing twice-daily measurements.

The trials were conducted with the ASV and shoreside systems over four days at Lake Waverley (figure 4.12), and a total of 47 mission runs were recorded. During trials, the effective Wi-Fi range of the shoreside router was identified to be between 50-60 m. This interfered with the replan recourse action for the long range mission, resulting in the supervisor initiating the return home recourse action if it was out of contact with the shoreside planner when δ_p was reached.

Figure 4.12: The ASV underway on a waypoint following task in Lake Waverley. Shoreside system setup in background.

Significant and unmeasured local gusts occurred during the course of the trials, which affected the control of the vehicle (see figure 4.13 as an example). Weather data from Bureau of Meteorology (2018) provides two reference measurements for the wind behaviour for each day of the trials. Additionally, the magnetic interference from the DC motors driving the azimuth thrusters would intermittently affect the magnetometer sensors within the IMU. This would lead to the vehicle's guidance system calculating incorrect LOS heading errors, and in the worst case causing the vehicle to circle a waypoint indefinitely. These challenges were overcome by adjusting the K_i gains of the PID controllers to be more aggressive, and by relocating the IMU towards the GPS mast. To correct the compass misalignment fault the mission had to be paused and manual override commands were given to the vehicle to align its heading with an external measurement of magnetic north so that the navigation system's compass datum could be reset. The effects of both the wind and the magnetic interference will also be important in section 4.5.2.

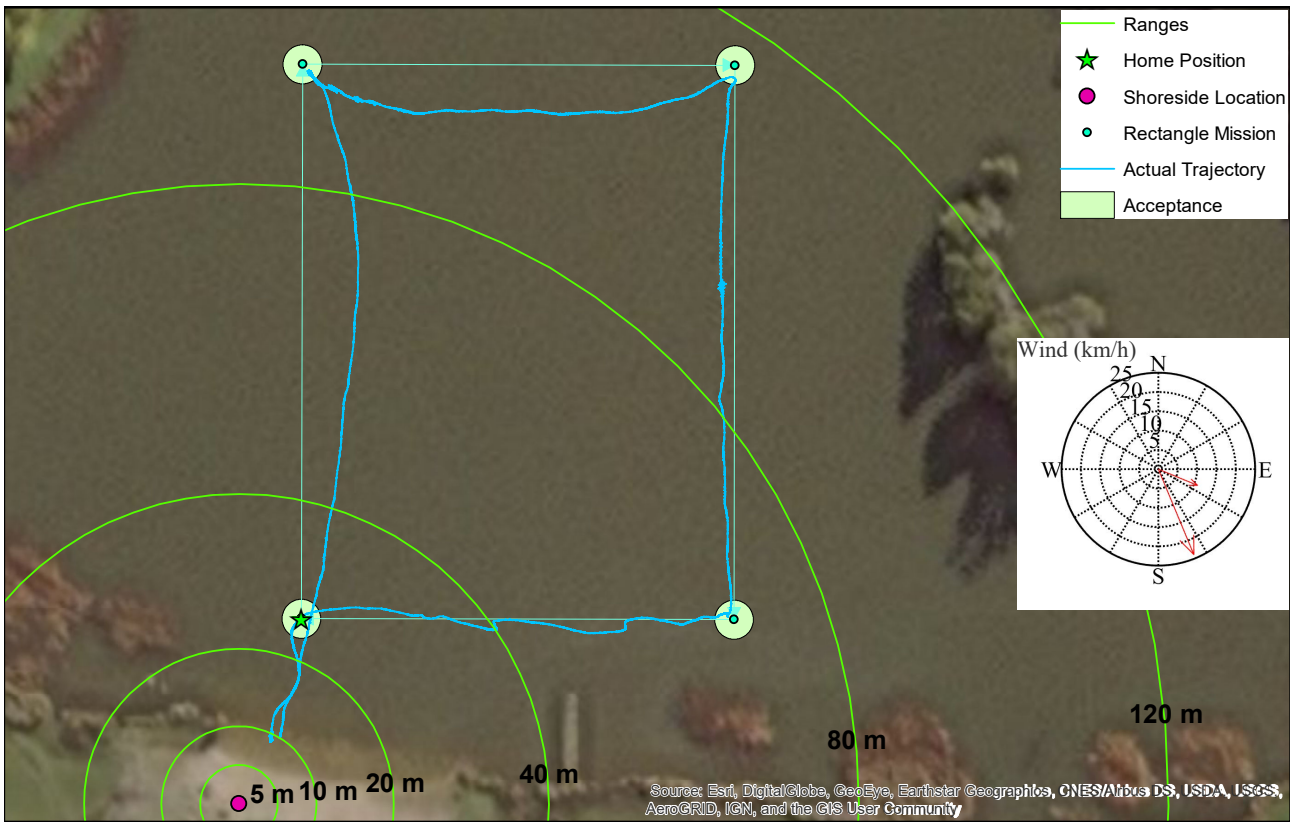


Figure 4.13: One of the rectangle missions performed by the ASV. Strong and intermittent North-Westerly gusts blew the vehicle sideways from the expected trajectory, leading to trajectories with cross-track errors that are sometimes large and sometimes small. The autopilot controller does not control motion in the sway direction, leading to arcs in the achieved trajectory.

4.5.1 Recourse Action Effects

Adjusting the δ thresholds for each of the survival functions produces significantly different behaviours from the supervisor. Combinations of different thresholds result in behaviours where it is difficult to identify which threshold contributes what to the result. To see the direct effects of δ_p and δ_t as they operate independently, this section looks at two runs of the medium size mission: one where δ_p is high and δ_t is off (i.e. < 0), and one where δ_t is high and δ_p is off.

Figure 4.14 presents a run of the medium size mission with $\delta_t = -1\text{e}10^{-9}$ and $\delta_p = 0.85$. From the trajectory of the ASV, it appears that the ASV attempts to finish the waypoint tasks in the order specified by the planner. When the threshold is crossed, the ASV performs a hold position manoeuvre and requests a new plan from the shoreside planner. The new plan budgets in one waypoint task before sending the vehicle to the home position. This behaviour is similar to the return home recourse actions described in Evers et al. (2014) and Shang et al. (2016), but is capable of planning in tasks that are along the route home.

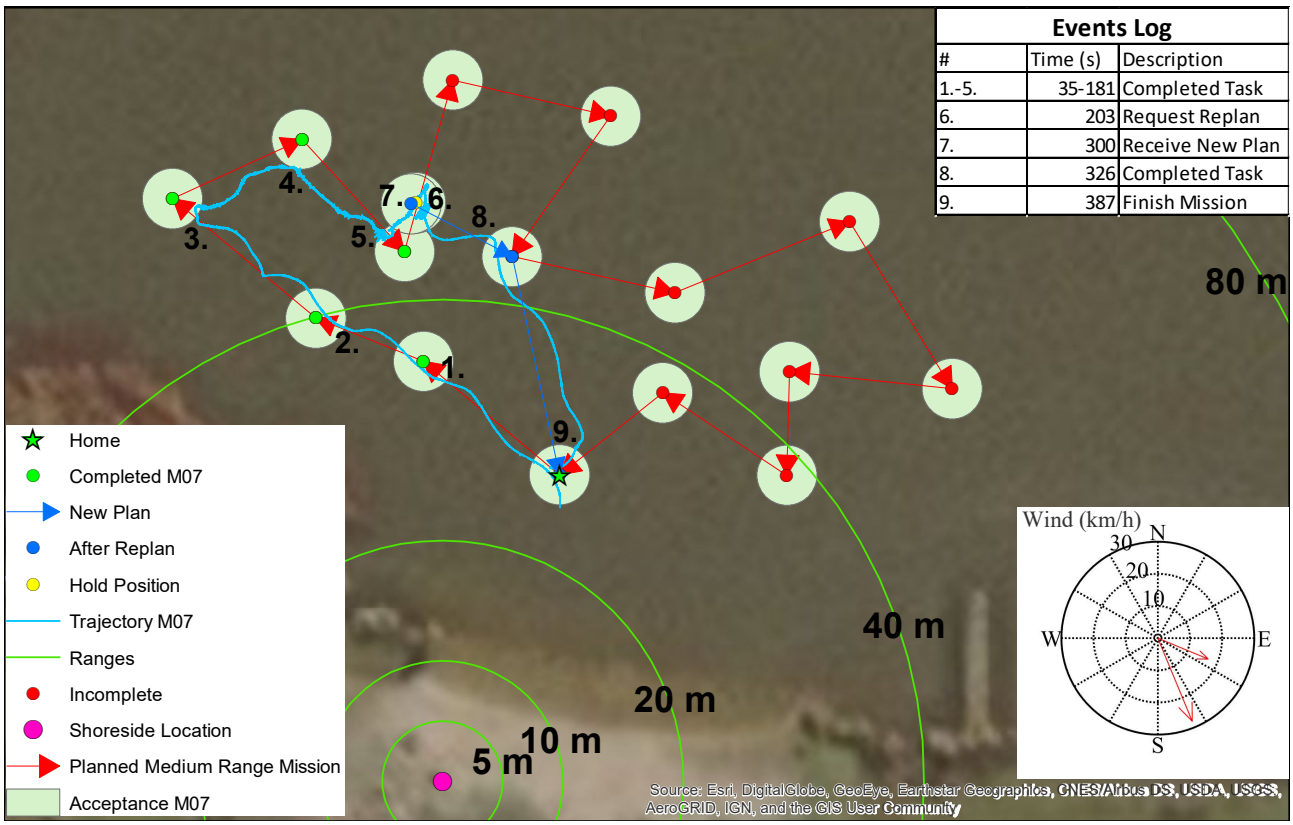


Figure 4.14: Progression of medium range mission 7. $\delta_p = 0.85$ and $\delta_t = -1e10^{-9}$.

Figure 4.15 presents the corresponding time series of battery, plan, and task survival function data for medium mission 7. The vehicle fell out of contact with the mission planner after it had successfully made a replan request, leading to a long hold position action. Once contact was reestablished, the vehicle proceeded on the new mission and completed it without triggering a new replan recourse action. The task survival function has drastic falling edges after a few seconds of performing a new task. This indicates that the planner is underestimating the energy cost of performing tasks, and also that the calculated H_t distributions have a small variance.

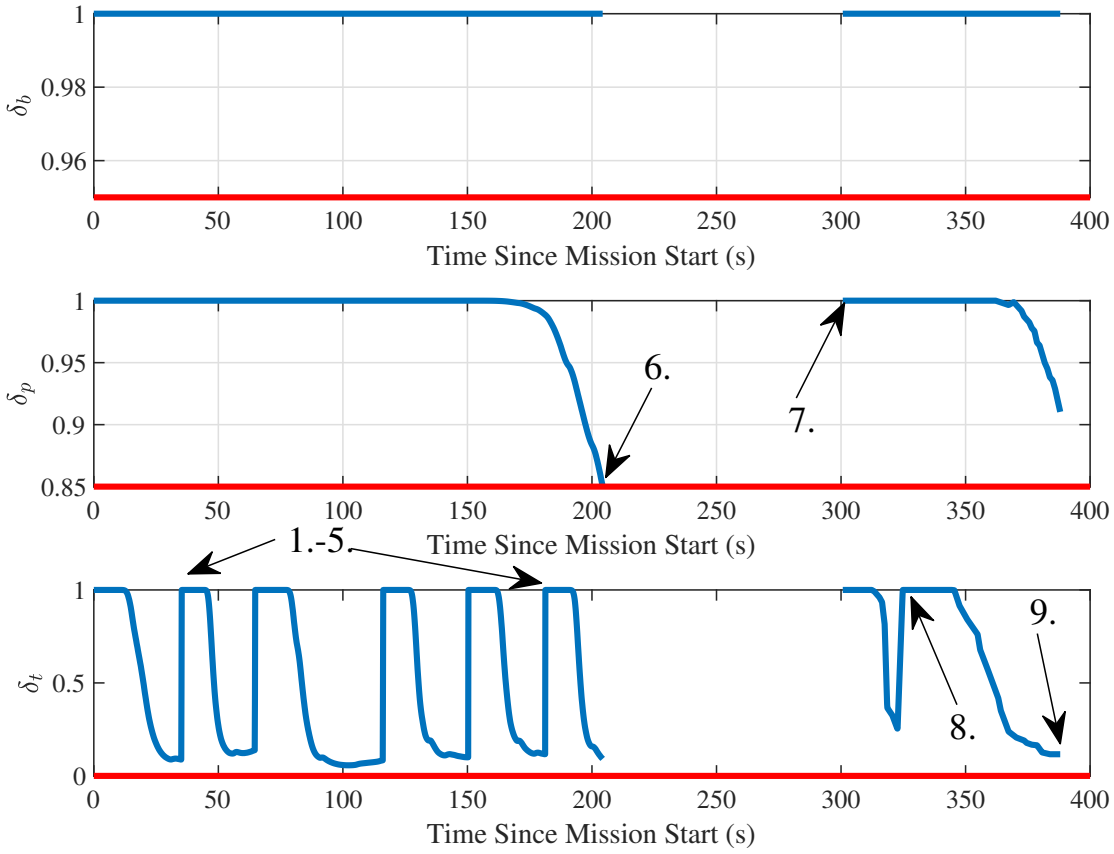


Figure 4.15: Survival functions of medium range mission 7. The numbered arrows correspond to the events log in figure 4.14. Task survival (δ_t) indicates that the generated plan is underestimating the power consumption of the vehicle. The gap in time (between 200-300 s) is because the survival functions are not considered by the supervisor while holding position.

As a basis of comparison, figure 4.16 presents run 22 of the medium range mission where $\delta_t = 0.5$ and $\delta_p = -1e10^{-9}$. The trajectory shows that the supervisor has a very short tolerance on pursuing a task before skipping it and moving on to the next. The vehicle was only able to complete tasks where its distance ended up being significantly closer than what was originally planned for it by virtue of the task skip heuristic.

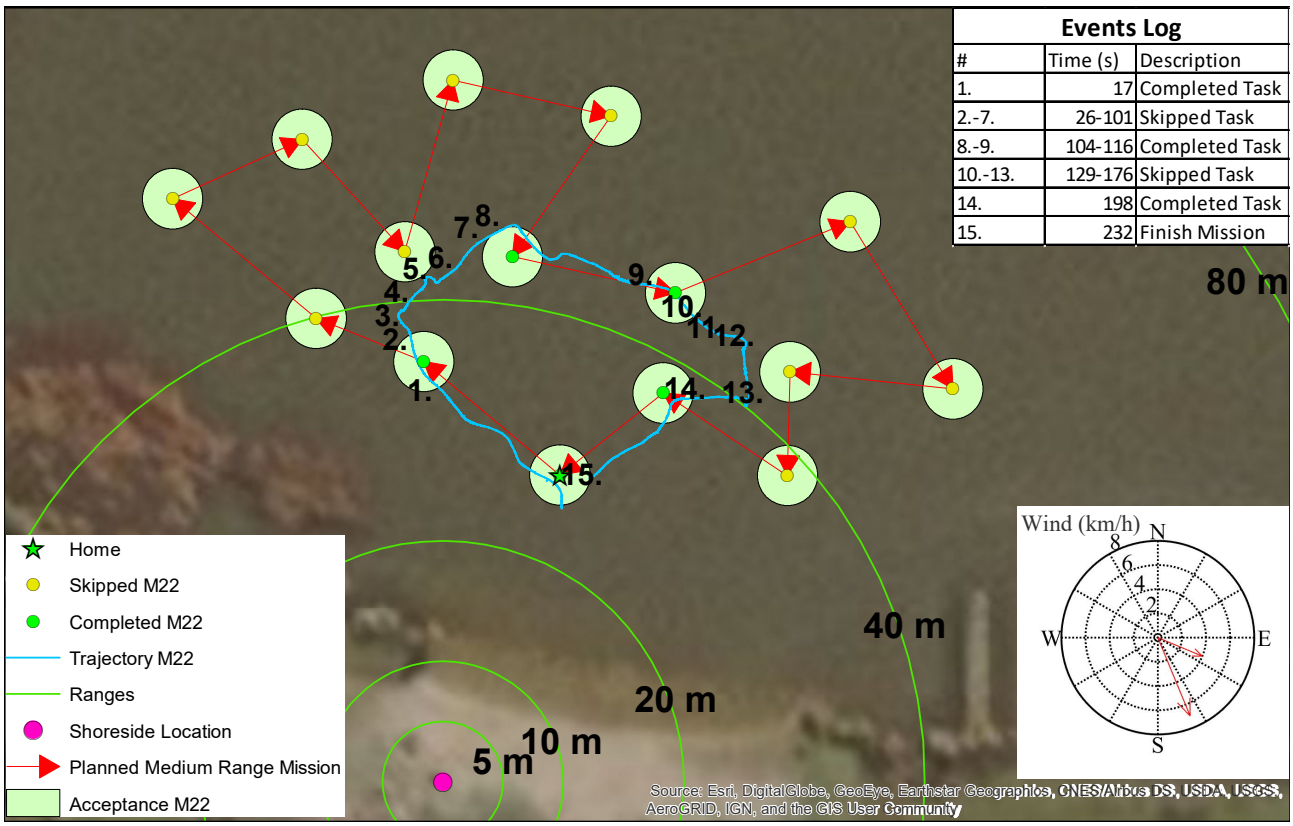


Figure 4.16: Medium range mission 22 progression. $\delta_t = 0.5$ and $\delta_p = -1e10^{-9}$.

The survival functions for mission 22 are slightly different from mission 7. Figure 4.17 shows that the task survival function began decaying almost immediately after the task was initiated. This is likely due to the more aggressive PID tunings and control output limits that were used on this run compared to mission 7. The controller outputted higher thrust commands to the thrusters, which results in much higher power consumption. The plan survival function also finishes quite close to 0.5, indicating that the vehicle finished the mission close to the expected energy consumption of the plan.

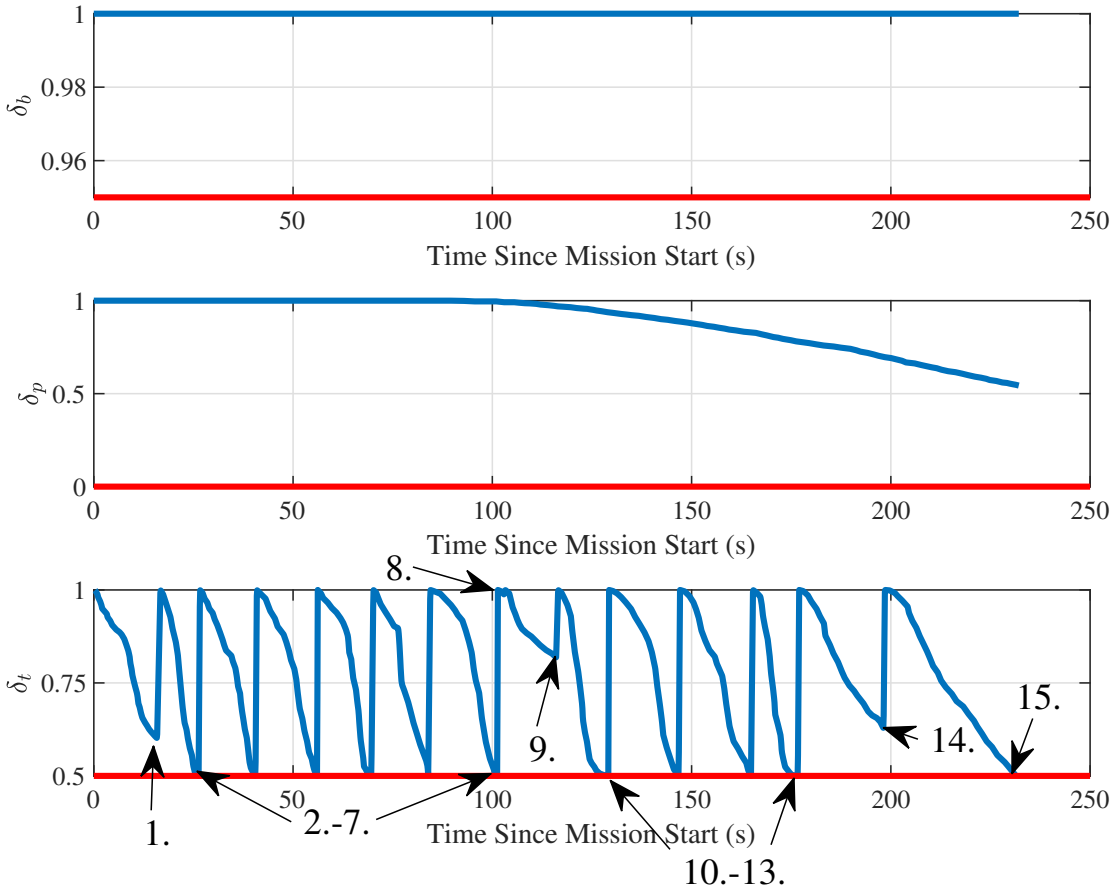


Figure 4.17: Survival functions of medium range mission 22. The numbered arrows correspond to the events log in figure 4.16. Task survival (δ_t) indicates that the generated plan is underestimating the power consumption of the vehicle.

Comparing the above missions, it appears that both strategies are regulating the total energy consumption of the vehicle with respect to the plan energy consumption expectation. In mission 7, the high δ_p threshold means that the supervisor makes conservative responses (i.e. it keeps to the schedule, and returns home early, visiting tasks along the way). Mission 22 shows that a high δ_t results in very quick jumps ahead in the schedule, leading to a more rounded coverage of the mission area. In either case, the results could have been improved with energy estimations that also included environment effects.

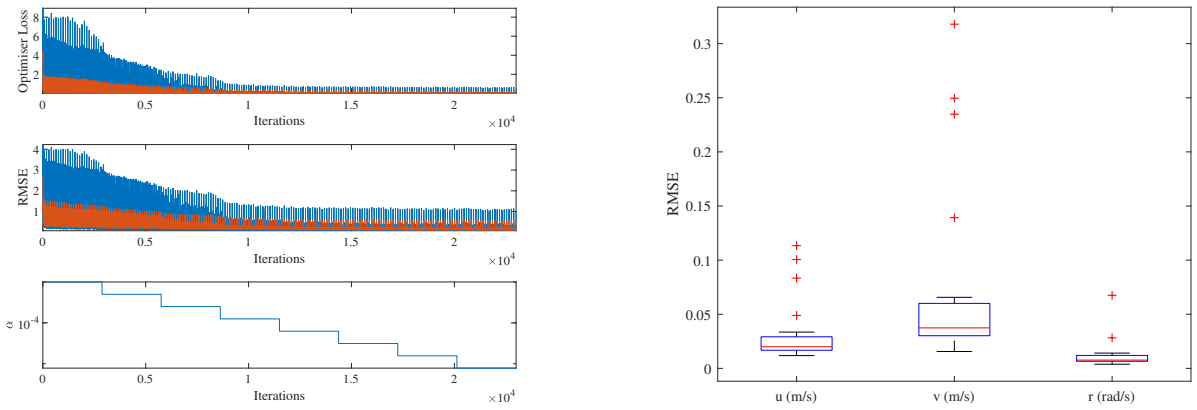
4.5.2 Forecaster Evaluation

Training

The data collected during the Waverley trials was used to train the hybrid energy forecaster model. During each mission the ASV switches between several operating modes: manual RC, autopilot, dynamic positioning, and idle. The power prediction LSTM was trained on data

from each operating mode, whereas the $\boldsymbol{\nu}$ prediction LSTM was trained on the subset of the data where the vehicle was operating in autopilot mode. The choice to only evaluate the $\boldsymbol{\nu}$ LSTM on only the autopilot data was made to simplify the controller component of the hybrid energy forecaster model. Forecasting the control input of an operator's RC command is counterintuitive, and forecasting the power consumption of a vehicle in idle mode simplifies to forecasting the hotel load. The controller component could be expanded to include the dynamic positioning controller in the future.

The data was split 9:1 between training and test data sets. Figure 4.18a presents the training curve for the $\boldsymbol{\nu}$ prediction LSTM. Figure 4.18b presents the Root-Mean-Squared-Error of the trained $\boldsymbol{\nu}$ prediction LSTM in performing one-step predictions on the test data sets. The performance on the test set indicates that the error in position between the predicted and actual will grow due to the integration of the velocity error.



(a) $\boldsymbol{\nu}$ Training Curve.

(b) $\boldsymbol{\nu}$ Test RMSE Results.

Figure 4.18: Training performance of the $\boldsymbol{\nu}$ prediction LSTM (4.18a) and the one-step prediction performance of the trained $\boldsymbol{\nu}$ prediction LSTM on the test data set (4.18b). The top and middle plots of 4.18a are the optimisation loss and standardised RMSE (including a red moving mean trend). The bottom plot is the learning rate, which is progressively halved during training.

4.5.3 Forecasting Kinematic State

To evaluate the forecast effectiveness, the trained predictor is first 'charged' with feed-in data from the test set. The length of the feed-in data does effect the quality of the forecast as the LSTM will have more valid historical data available, but a trade-off has to be made with computational resources (i.e. more memory is needed to store longer feed-in). The first 5 seconds of each of the test input sets were reserved for feed-in (we found 5 seconds to be sufficiently long enough for the forecaster to produce accurate forecasts).

Figure 4.19 presents the error in distance-to-target between the actual and forecast for each autopilot task (each coloured line representing a task). For positive $D(t) - \hat{D}(t)$, the forecasted vehicle position is closer to the target than actual (optimistic forecast) and *vice versa* for negative $D(t) - \hat{D}(t)$ (pessimistic forecast). The error in $\boldsymbol{\nu}$ between actual and forecast is integrated, which increases the rate of error growth in forecasted $\boldsymbol{\eta}$. This in turn affects the guidance module, which will use a different distance-to-target to calculate the LOS heading error, $\psi_e(t)$ (displayed in figure 4.20).

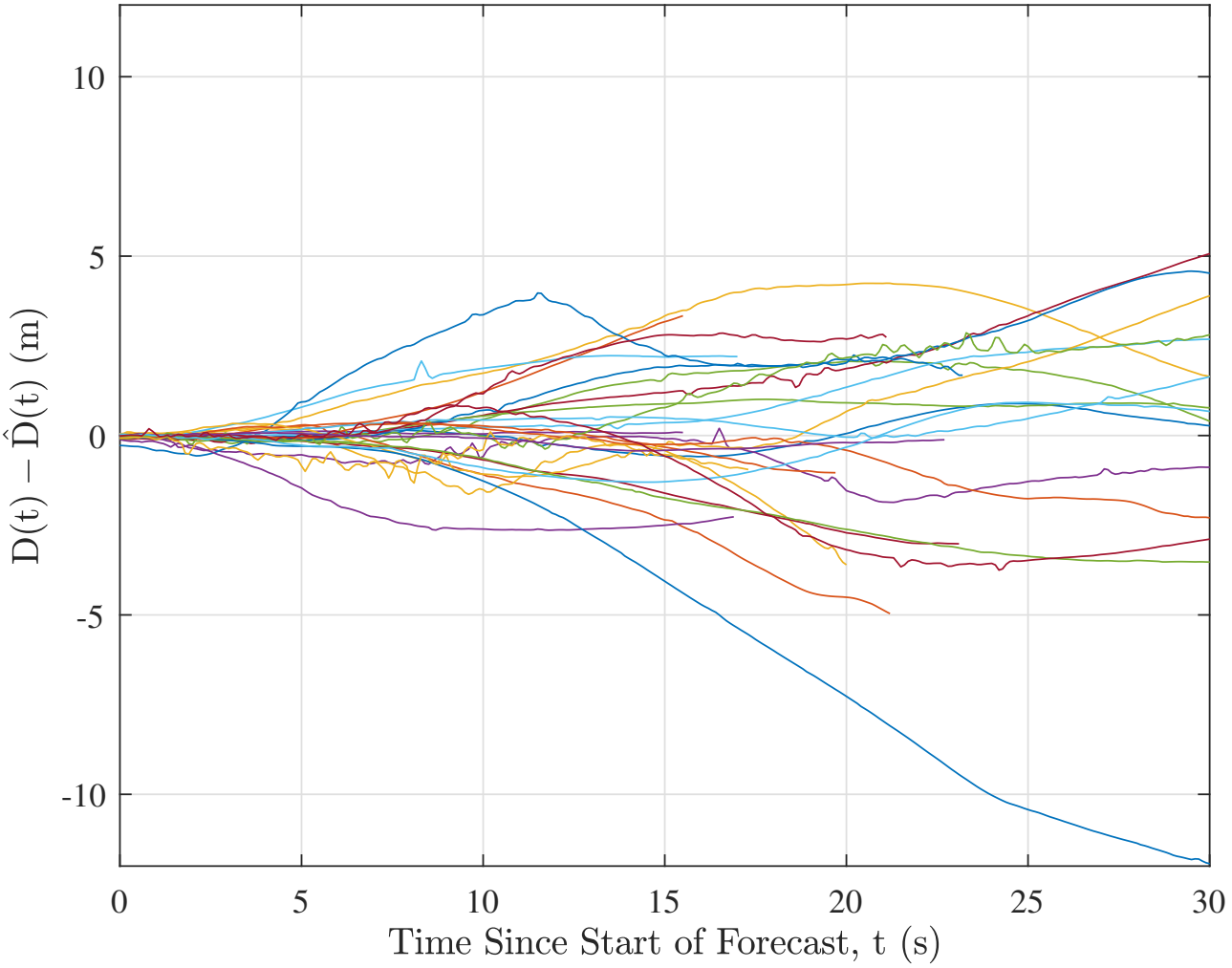


Figure 4.19: Error between actual and forecast distance to target over a 30 s window. The $\boldsymbol{\nu}$ LSTM and following autopilot controller were fed with 5 seconds of ground truth data before forecasting began.

In general, both the distance-to-target and the LOS heading error between forecasted and actual are reasonably small for the first 5 seconds of forecasting. Some of the significant errors in figure 4.20 are because of strong wind gusts that occurred after forecasting began (i.e. the feed-in data did not indicate enough information about the wind effects to the LSTM). In this respect we suggest extending the kinematic LSTM by including wind speed and direction

measurements as additional input variables. This will allow the trained LSTM to learn a model for the estimation of τ_{wind}, improving the accuracy of the prediction. Including the next-step wind measurements as output variable would allow the LSTM to make forecasts on future wind speed and direction. Reliable and frequent wind speed and direction measurements must be obtained through an on-board anemometer for this approach to be possible (data from weather stations may be low-frequency or not entirely representative of the local conditions). Although the training and test data were filtered for any run segments that contained identifiable magnetic interference to the IMU (described in the introduction of section 4.5), imperceptibly small compass errors cannot be discounted from the data and may not be consistent enough for the LSTM to incorporate.

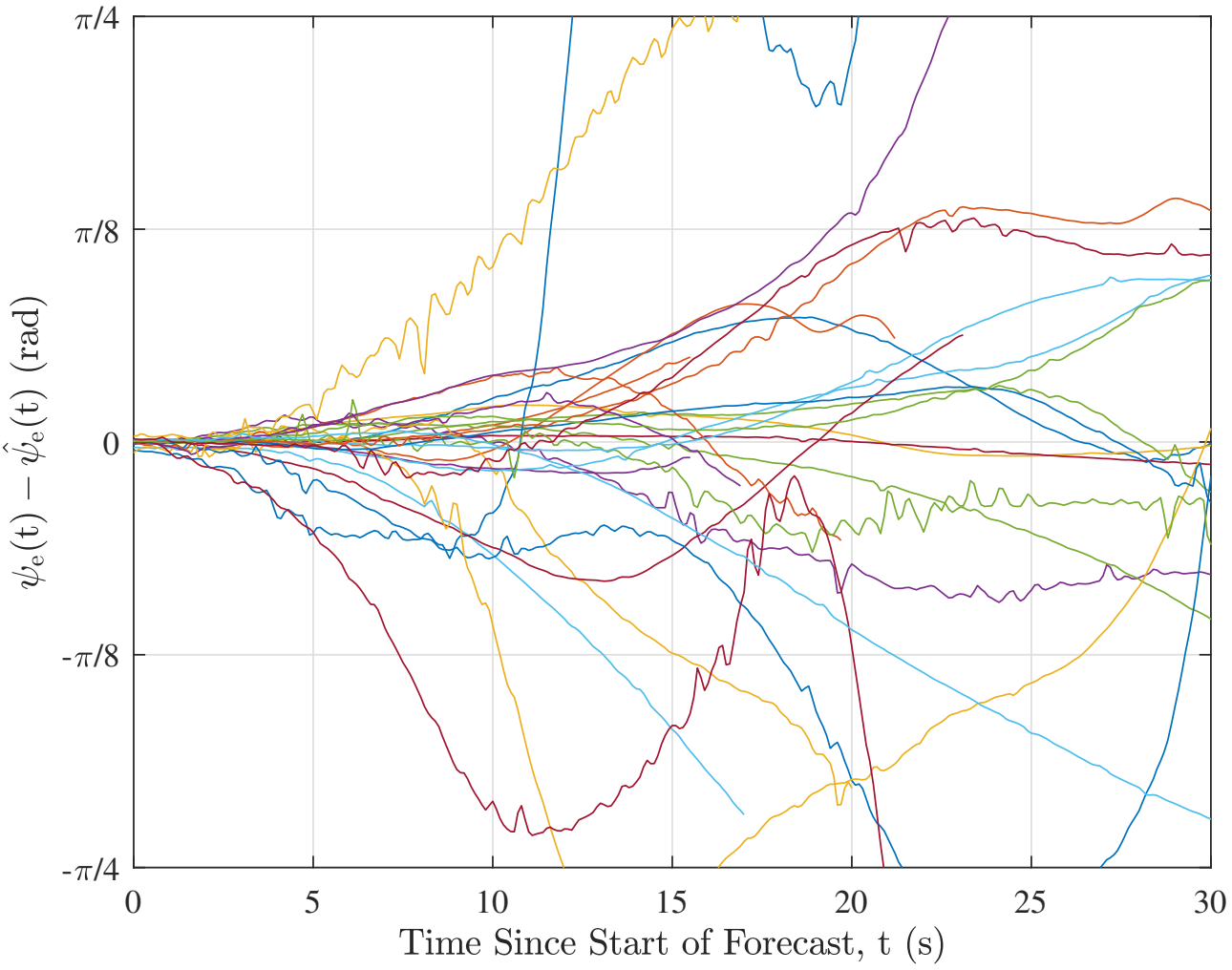
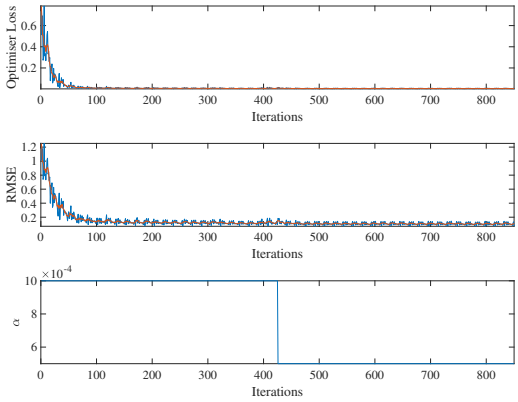


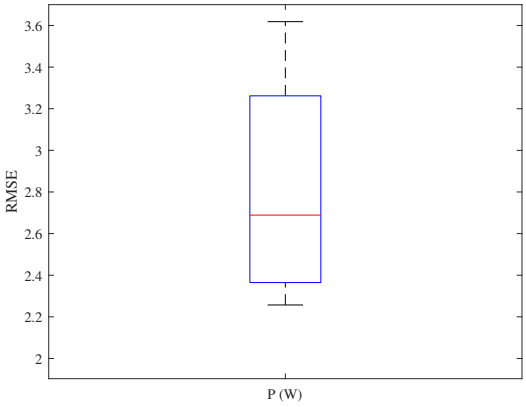
Figure 4.20: Error between actual and forecast LOS heading error. The $\boldsymbol{\nu}$ LSTM and following autopilot controller were fed with 5 seconds of ground truth data before forecasting began.

4.5.4 Forecasting Energy

Figure 4.21a presents the training curve for the power prediction LSTM. Figure 4.21b presents the Root-Mean-Squared-Error of the trained power prediction LSTM in performing one-step predictions on the test data sets. The power prediction timeseries for the actual and forecast are integrated to obtain the energy aggregate for each task, which is used for the rest of the forecast performance investigation.



(a) Power training curve with optimiser loss (top), RMSE (middle) and learning rate α (bottom).



(b) Box and whisker plot of the Root-Mean-Squared-Error between the test data and the power one-step prediction of the LSTM.

Figure 4.21: Training and one-step prediction test results for the power prediction LSTM.

As soon as the forecast begins there is an error between the actual and forecast. To properly evaluate the forecaster, some measurement for determining when the error is too large to be reliable (i.e. the reliable forecast horizon) is required. A suitable candidate is the confidence interval of the planner-generated energy distribution as this is based on what the Monte Carlo simulation and Gaussian approximation produces for the given task. The 95% and 50% confidence offsets (i.e. $1.96 \pm \sigma(k)$ and $0.67 \pm \sigma(k)$) were chosen as standard reliability thresholds for the forecaster. The confidence offsets were calculated by obtaining E_σ for a given M_o and plotting it with the total expected duration of the task transition. Figure 4.22 shows the 95% ($1.96E_\sigma$) and 50% ($0.67E_\sigma$) confidence offsets of the distributions for the Monte Carlo simulated energy calculation of the all the potential tasks in the midsize mission from section 4.5.1.

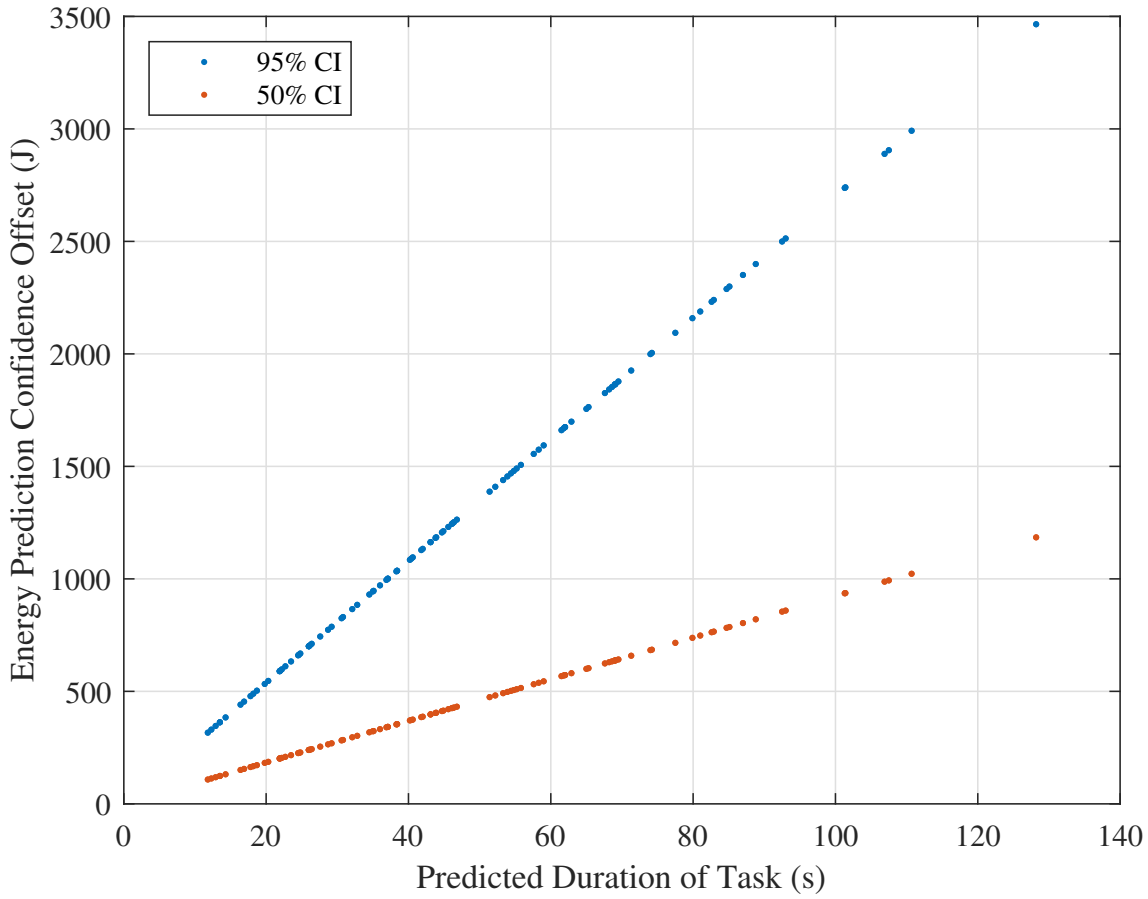


Figure 4.22: 95% and 50% confidence offsets of the predicted energy distribution for all possible tasks that are available in the midsize mission. Distributions were calculated from 10,000 simulations of the vehicle dynamics model.

As can be seen from the growth of the confidence intervals, the standard deviation of the approximated Gaussian distribution for each autopilot manoeuvring task increases linearly with the duration. This is because each trajectory that the planner is considering for each task is a straight line and assumes constant forward velocity of the vehicle. For missions where there are obstacles that cause the trajectory to be curved, or that have changes in velocity the linear trend will not be as clear.

The lines that fit the 95% and 50% confidence offsets show how the confidence of the Monte Carlo simulation is expected to decrease (spread) as the task length becomes longer. By comparing the spread of these confidence lines with the prediction interval spread of the forecast error, the time at which the forecaster prediction confidence exceeds the expected confidence can be identified. This time is indicative of when the forecast becomes less reliable than the model prediction made by the mission planner and is our definition of the reliable forecast horizon.

The full hybrid energy forecaster (see figure 4.10) was evaluated on test data using autopilot "move to position" task data only (28 different tasks in total). The $\boldsymbol{\nu}$ and power prediction LSTMs were 'charged' with a feed-in of the ground truth data. The forecaster was evaluated over several trials with feed-in lengths increasing in 1 s increments from 1 s to 15 s. Figure 4.23 below presents the error between the actual energy consumption and the forecast energy consumption with a feed-in length of 5 s.

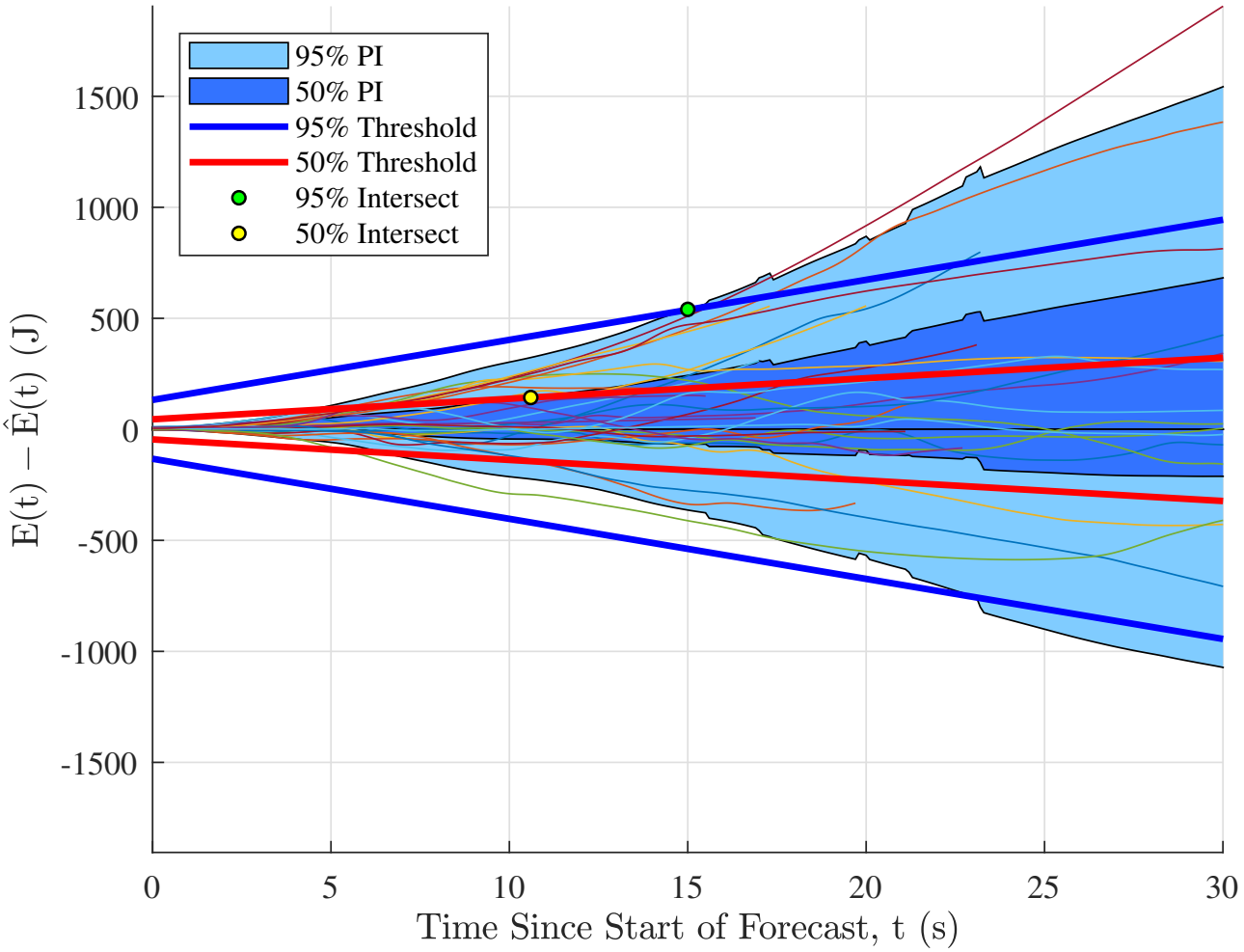


Figure 4.23: Error between actual and forecast energy consumption. Power prediction LSTM was provided with 5 s of ground truth power/duty cycle data before forecasting began.

Each run of the test data is represented by the thin coloured lines. The lighter shaded area is the 95% confidence interval of the forecast, and the darker shaded area is the 50% confidence interval of the forecast. What can be seen immediately from figure 4.23 is that the error is biased towards the positive and increases over time. This means that the forecaster has a higher chance of forecasting an underestimate of the required energy for any horizon. The green marker indicates when 95% of the energy error has reached the 95% confidence interval of the modelled energy prediction from the planner. The yellow marker indicates the same for

a 50% confidence interval intersection. A reliable forecast horizon can be determined through these intersections. For example, if the criteria for a reliable forecast horizon was to be 95% sure that the forecaster error is within 95% of the predicted energy distribution, then the time at the green marker (approximately 15 s) would be an appropriate horizon.

A slightly more conservative approach would be to consider the absolute energy error as in figure 4.24. This brings the point at which it is 50% certain that the forecaster error is within 50% of the predicted energy distribution forward to be earlier in time than what is calculated in figure 4.23. However information about whether the forecaster is optimistic or pessimistic is lost as the absolute energy error always assumes that the forecaster is optimistic. Additionally, the point at which it is 95% certain that the forecaster error is within 95% of the predicted energy distribution is pushed back in time compared to figure 4.23. For these reasons we recommend assessing the reliability of the forecaster according to the positive/negative energy error data as in figure 4.23.

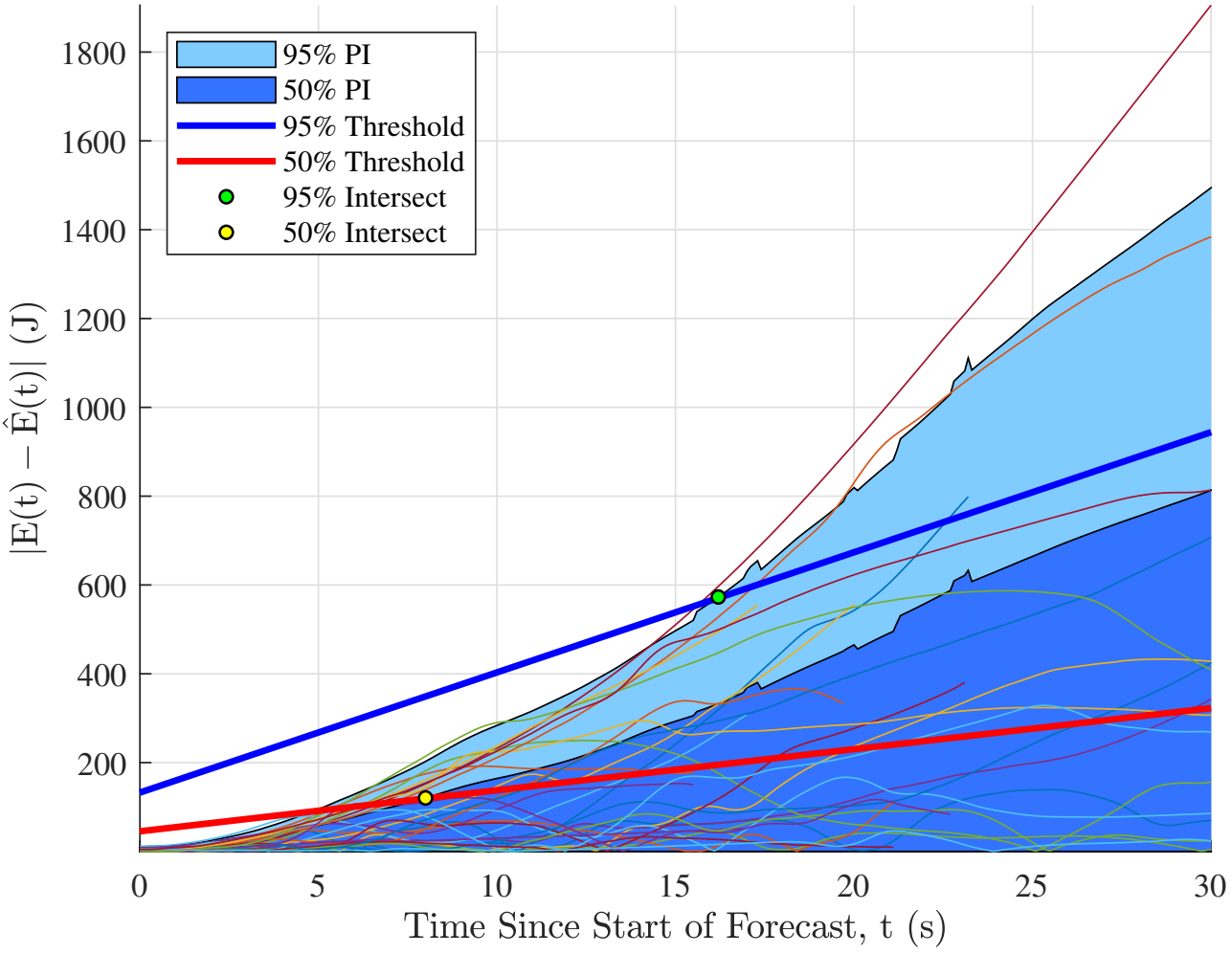


Figure 4.24: Absolute error between actual and forecast energy consumption. Power prediction LSTM was provided with 5 s of ground truth power/duty cycle data before forecasting began.

4.6 Discussion and Recommendations

Three components of energy-based planning have been described: the development of energy planning for AMVs as an extension of researched solutions to the OP-SW; the development of an ASV system capable of realising and testing the two-stage energy planning approach; and *post hoc* analysis of the vehicle's kinematic response and energy consumption with the goal of developing an energy forecasting solution. These three components contribute to the goal of producing a mission planner that can safely plan under uncertainty, can deal with communication dropouts between the vehicle and the operator, and can reliably forecast future demand for the purposes of online mission supervision and decision-making for the AMV.

In section 4.5.1, we presented two different runs of the same mission plan where the supervisor was tuned with different survival thresholds. In the first run, the supervisor ignored the realisations of each task leg and focussed on the total aggregated energy. The ASV stayed on its predetermined mission plan until the supervisor was 85% sure that its total consumed energy for the mission had not exceeded the predicted energy consumption. At this point the ASV deviated from its expected plan and requested a new plan from the shoreside system (notifying the operator in the process). The supervisor demonstrated a conservative policy, making it easier for the operator to be sure of what the vehicle was doing (as it was likely to be doing what was initially determined by the first stage planner). From a geographical standpoint, (i.e. if this test mission's goal was to survey an area), the operator can only be confident that the Western half of the area was adequately surveyed.

In the second run, the supervisor was tuned with survival thresholds that focussed on the realisations of the task leg and ignored the total aggregated energy. This resulted in the ASV only visiting targets that were achieved **before** the supervisor was more than 50% sure that the energy consumed for the task was greater than the predicted. Tasks that took longer than expected were skipped, making this version of the supervisor exhibit opportunistic behaviour. From the operator's perspective, it is a lot harder to predict if the supervisor will skip a task or complete it. However, the surveyed area is much more evenly distributed around the vehicle's rendezvous point.

There are many other potential behaviours that could be exhibited by the supervisor by simply using different combinations of survival thresholds. Given the permutations, observing how each combination will affect the supervisor's reaction to the realised environment in an exhaustive fashion isn't feasible within the practical limitations of physical field experiments. The next step

for evaluating this planner will be through simulation, using OP-SW testing data to properly quantify the mission outcomes (i.e. how many tasks were completed, did the vehicle return home within energy constraints, etc.) for many combinations of battery, plan, and task survival thresholds.

The purpose of the field trials was to evaluate how the planner behaves when exposed to unaccounted-for uncertainty (e.g. the planner was unaware of the effects of wind on the power consumption of the vehicle). Future field testing of the mission planner and ASV should take into account measurements of the environmental factors and the vehicle model uncertainty. Properly identifying the vehicle model means system identification of the vehicle's hydrodynamic and aerodynamic properties through captive model testing, free running experiments (preferably with controlled wind effects), or computational fluid dynamics simulation of the loads on the ASV's submerged hull (hydrodynamics) and superstructure (aerodynamics). This would allow for the design of an appropriate marine vehicle model for use in an EKF, resulting in better kinematic state estimations. It would also allow for the design of a better control process through simulation of the fully identified vehicle kinetics model. Redesign of the vehicle's IMU location will also improve the reliability of heading estimation by reducing magnetic interference.

The ultimate goal is for the supervisor to predict the survival probability and make active changes to the plan that will positively affect its progress towards completing missions. The data-driven approach of the LSTM forecaster seems promising, even with the noisy kinematic state and energy consumption training data used. The forecaster was shown to have a reliable horizon between 5 and 15 seconds, depending on the operator's criteria. This horizon is too small for it to be of immediate practical use for this prototype model, as the data that it was trained on originated from noisy, low-cost components. However, being able to accurately forecast energy consumption ahead of time *in situ* presents the vehicle planner with the opportunity to make energy-informed decisions ahead of time, which could be critical in ensuring the vehicle returns to a recoverable position.

Noise and inteference effects on the navigation sensors (such as magnetic interference on the IMU) could not be fully accounted for and comes with the territory of using low-cost sensors. We suggest that the reliable energy forecast horizon could be extended if the LSTM was trained on data from commercially available state-of-the-art navigation equipment. As a separate entity the energy forecaster could be implemented on many different vehicle types, requiring reliable previous mission history data. Transfer learning shows promise for enabling LSTMs trained

on one vehicle data set to be adapted to another vehicle without significant loss in accuracy (Laptev et al., 2018).

The mission planner energy model could be improved by including more information from the environment or environment forecasts (such as wind, waves, currents and traffic conditions), such as what has been achieved in Hollinger et al. (2016). Additionally, the current trajectory generator assumes the vehicle instantly changes orientation to face the next task. The trajectory generator could be developed further to more realistically represent the vehicle's true path over ground.

4.7 Conclusions

This chapter addresses two challenges for marine vehicle mission planning: adapting a plan based on the realisation of a partially unknown environment, and forecasting the progress of a given plan based on uncertain environment observations. In both cases, quantifying the environment as much as possible reduces uncertainty and allows for more reliable plan adaptation and plan forecasting. The power supply of the vehicle is a critical point of mission failure (Brito et al., 2014), thus energy consumption was chosen as the mechanism for evaluating the environment and the vehicle's progression along the plan.

Making reliable forecasts of the vehicle's energy consumption improves the decision-making agent's skill in generating mission critical decisions ahead of time. Most importantly this reduces the risk of the vehicle getting into a situation where there are no available remedies. It also means that the vehicle can use the time saved to do more tasks.

The results presented in this chapter demonstrate the potential that energy planning has for quantifying the uncertainty in mission planning for marine platforms. An existing energy-aware mission planner was expanded into a two-stage planner and was tested on a custom AMV system operating in an uncertain environment. The second-stage supervisor agent uses survival functions as decision-making criteria for recourse actions, which the operator can threshold to control the supervisor's recourse policy. The energy forecaster demonstrated that predicting the energy consumption of a marine vehicle is possible even with an incomplete representation of the marine environment. Combining these three components produces a mission planner capable of creating an energy-aware plan; that is able to forecast the feasibility of the vehicle's current plan into the future; and take preemptive action to minimise the risk of power supply failures.

References

Ai-Chang, M., Bresina, J., Charest, L., Chase, A., Hsu, J. C. J., Jonsson, A., Kanefsky, B., Morris, P., Rajan, K., Yglesias, J., Chafin, B. G., Dias, W. C., and Maldague, P. F. (2004). MAPGEN: mixed-initiative planning and scheduling for the Mars Exploration Rover mission. *IEEE Intelligent Systems*, 19(1):8–12.

Brito, M., Smeed, D., and Griffiths, G. (2014). Underwater glider reliability and implications for survey design. *Journal of Atmospheric and Oceanic Technology*, 31(12):2858–2870.

Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23.

Bureau of Meteorology (2018). Launceston, Tasmania November 2018 Daily Weather Observations. BoM, http://www.bom.gov.au/climate/dwo/201811/html/IDCJDW7025.201811.shtml. Accessed: December 2018.

Caley, J. A., Lawrance, N. R. J., and Hollinger, G. A. (2017). Deep networks with confidence bounds for robotic information gathering.

Chao, I., Golden, B. L., and Wasil, E. A. (1996). The team orienteering problem. *European Journal of Operational Research*, 88(3):464–474.

Evers, L., Glorie, K., van der Ster, S., Barros, A. I., and Monsuur, H. (2014). A two-stage approach to the orienteering problem with stochastic weights. *Computers & Operations Research*, 43:248–260.

Ferreira, A. S., Costa, M., Py, F., Pinto, J., Silva, M. A., Nimmo-Smith, A., Johansen, T. A., de Sousa, J. B., and Rajan, K. (2018). Advancing multi-vehicle deployments in oceanographic field experiments. *Autonomous Robots*.

Fossen, T. (2011a). *Handbook of Marine Craft Hydrodynamics and Motion Control*, chapter 9, pages 232–233. John Wiley & Sons, Ltd.

Fossen, T. (2011b). *Handbook of Marine Craft Hydrodynamics and Motion Control*, chapter 10, pages 257–266. John Wiley & Sons, Ltd.

Fossen, T. (2011c). *Handbook of Marine Craft Hydrodynamics and Motion Control*, chapter 12, pages 398–415. John Wiley & Sons, Ltd.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2017).

LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, 28(10):2222–2232.

Hamza, A. and Ayanian, N. (2017). Forecasting battery state of charge for robot missions. In *Proceedings of the Symposium on Applied Computing*, SAC '17, pages 249–255, New York, NY, USA. ACM.

Hollinger, G. A., Pereira, A. A., Binney, J., Somers, T., and Sukhatme, G. S. (2016). Learning uncertainty in ocean current predictions for safe and reliable navigation of underwater vehicles. *Journal of Field Robotics*, 33(1):47–66.

Laptev, N., Yu, J., and Rajagopal, R. (2018). Reconstruction and Regression Loss for Time-Series Transfer Learning. In *Proceedings of ACM Conference (SIGKDD MiLeTS'18)*, London, UK. ACM. Poster.

McGann, C., Py, F., Rajan, K., Thomas, H., Henthorn, R., and McEwen, R. S. (2008). A deliberative architecture for AUV control. In *2008 IEEE International Conference on Robotics and Automation*, pages 1049–1054, Pasadena, CA, USA.

Moore, T. and Stouch, D. (2014). A generalized Extended Kalman Filter implementation for the robot operating system. In *Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13)*, Padua, Italy. Springer.

Py, F., Pinto, J., Silva, M. A., Johansen, T. A., Sousa, J., and Rajan, K. (2016). EUROPtus: A mixed-initiative controller for multi-vehicle oceanographic field experiments. In *International Symposium on Experimental Robotics*, pages 323–340, Tokyo, Japan.

Sadrpour, A., Jin, J. J., and Ulsoy, A. G. (2013). Mission energy prediction for unmanned ground vehicles using real-time measurements and prior knowledge. *Journal of Field Robotics*, 30(3):399–414.

Shang, K., Chan, F. T. S., Karungaru, S., Terada, K., Feng, Z., and Ke, L. (2016). Two-stage robust optimization for orienteering problem with stochastic weights. *arXiv e-prints*, page arXiv:1701.00090.

Sotzing, C. C., Evans, J., and Lane, D. M. (2007). A multi-agent architecture to increase coordination efficiency in multi-AUV operations. In *OCEANS 2007 - Europe*, pages 1–6, Aberdeen, Scotland, UK.

Thompson, F. and Galeazzi, R. (2018). Robust mission planning for autonomous marine vehicle fleets. *Manuscript submitted to: Engineering Applications of Artificial Intelligence*.

Tsiligirides, T. (1984). Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35(9):797–809.

Tsiogkas, N. and Lane, D. M. (2018). An evolutionary algorithm for online, resource constrained, multi-vehicle sensing mission planning. *arXiv e-prints*, page arXiv:1801.03552.

Zaytar, M. A. and El, C. (2016). Sequence to sequence weather forecasting with long short-term memory recurrent neural networks. *International Journal of Computer Applications*, 143:7–11.

Zlot, R. M. (2006). *An Auction-Based Approach to Complex Task Allocation for Multirobot Teams*. PhD thesis, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA.

Chapter 5

Conclusions

5.1 Summary

Mission planning for AMVs is challenging on several levels. In the pre-deployment stage the marine environment is only partially observable by the operator. This means that, when making a plan, potential risks must be first identified and then evaluated on their likelihood of occurring, and then contingency measures are designed as recourse for these risks. During the deployment stage, vehicles, especially in underwater operations, do not have constant communication with the operator once they are deployed. This means that they cannot rely upon an operator to arbitrate decisions at all times, and must be able to perform their own risk detection and arbitrate recourse actions *in situ*. Operators and vehicles are challenged to A) make a plan with enough flexibility to adapt to uncertain conditions, and B) successfully detect and adjust to the realisation of actual conditions.

A critical component of an AMV's survivability and reliability for a given mission is its energy supply (in most systems this is an electric battery). AMVs use their energy supply to move, to operate and log payload data, to communicate, to localise and navigate, and to process data for online decision-making. Failing to accommodate energy related risks into a plan increases the risk of jeopardising the mission, the consequences of which range from slight (e.g. the vehicle stopped using and recording its payload to conserve power) to severe (e.g. the vehicle powered down and could not be found).

In the current narrative of mission planning for AMVs, temporal resource constrained planning has seen significant success in providing the critical infrastructure needed to perform coordinated missions with AMVs (Ferreira et al., 2018). Planning is done first and foremost

by allocating and scheduling tasks to vehicles so that the total time taken for the mission is minimised. The energy supply of each vehicle is considered by the planner through a 'battery time' constraint that ensures no vehicle is given a schedule of tasks that it cannot complete. Uncertainty is considered first through specification of the upper and lower duration for each task. Adjustments to the plan are made through frequent updates of the onboard planner during deployment.

In the pursuit of reducing the uncertainty in determining the energy demands for task completion, this thesis has investigated the use of energy modelling of AMVs for an energy resource-constrained automated planner. Unlike temporal planning, energy planning considers the loadings overcome by the vehicle while completing tasks, resulting in energy demand estimates with better quantified uncertainty. An energy planning framework was constructed around this energy modelling approach, and was tested in both simulation and on a prototype ASV in a lake environment. During trial missions of the ASV, the task energy demand estimates obtained through energy modelling were used as a reference for the deployed vehicle to make decisions independently from the operator. An onboard mission energy supervisor agent was designed to use the task energy reference to calculate the probability of completing a task within budget, and was deployed on the ASV. From the results presented in chapter 4, it was demonstrated that the supervisor agent allows the operator to specify intuitive recourse thresholds that control its decision-making policy to exhibit conservative or venturesome energy spending behaviours.

The next section details the specific outcomes that have contributed to these findings, followed by a section considering the impact of the outcomes. Finally, this thesis is concluded with a set of recommendations for future research.

5.2 Outcomes

The aim of this thesis was: *"to develop and implement a new automated planner that increases the survivability and reliability of a deployed fleet of AMVs"*. The following points summarise the outcomes of the thesis in pursuit of this aim:

1. Reviewed the state-of-the-art of mission planning for AMV fleets and revealed a need for a more complete representation of the vehicles' interactions with the environment is needed to increase the reliability of planning prediction. Energy consumption estimation through the hydrodynamic modelling of vehicle was proposed as a better representaion than the purely temporal approach of existing mission planners.

2. A robust mission planner for AMV fleets was designed with energy as the base planning resource. The system decomposes tasks into actions based on the Team Orienteering Problem, and uses Monte Carlo simulation to obtain energy consumption distributions for the vehicles to complete tasks.

3. Simulation and field experiments have been conducted to evaluate the performance of the planner. In the simulations the planner was shown to decompose large missions into subset planning problems, which are then solved for near-optimum plans.

4. In the field experiments, a supervisor agent designed to use survival functions of the provided task energy distributions was demonstrated to produce conservative and opportunistic behaviours based on threshold parameters.

5. In review of the collected trajectory and energy consumption data, an energy forecaster was designed to allow the supervisor to look ahead in time and make critical decisions earlier. The forecaster was shown to have a reliable planning horizon of 10 seconds.

5.3 Impact

The core of this thesis advocates energy planning as a viable avenue for producing plans that include a better quantified representation of uncertainty than what is currently available in temporal planning. Previously, energy planning was done at the trajectory planning level, but to the author's knowledge has not been considered at a higher planning and scheduling level. Considering a mission plan in terms of energy rather than time changes the perception of the problem significantly. Tasks that are equidistant from each other in time cost may not be equidistant when represented as energy cost. In this respect, energy planning presents the opportunity for operators to make energy efficient use of their vehicles, allowing them to keep their vehicles in the water for longer and with a better understanding and control of the risks. This links back to the discussion on cost-effectiveness at the beginning of this thesis (figure 1.1). Through the quantification of the vehicle's energy consumption, the mission planner proposed in this thesis is able to automatically create and monitor plans that balance the productivity of the vehicle with its safety.

Necessary input from the operator is minimal. Through the specification of three survival activation thresholds, the operator is able to control the policy of the vehicle when it encounters uncertainty in the mission. This is a novel way for the operator to specify the safety margins

that control the vehicle's exposure to risk. The operator is less involved with the computation of fail-safe metrics and margins, but is able to specify the confidence limits of the supervisor agent. Through this hands-off approach the level of risk that the vehicle is exposed to in future missions can be tuned to minimise the number of deployments and retrievals in an expedition, increasing cost-effectiveness (as in figure 1.1d).

In section 3.2.6, a method for quickly decomposing a mission that was too large into a set of feasible sub-missions was presented. This method allows an operator to determine (from an expedition planning perspective) the expected schedule based on the number and locations of required vehicle deployments. From an operations research point of view it prunes many edges in the TOP that would not be feasible due to the range limitations of the provided vehicles, reducing the search space and solving time. This method could be extended into many other problem domains in OR.

Specific to particle swarm optimisation, section 3.2.8 improves the initial search size of the swarm, increasing the chances of it finding good solutions earlier in the solving process. Particles that have had a history of weak solutions are pruned, which was demonstrated to save up to 20 seconds on total computation time (with no discernable drop in solution quality) for large swarm populations (see figure 3.9). This method could be adopted and extended into other PSO algorithms, suitable for problems where the score space is sufficiently smooth (although further verification is required).

In section 4.2, the two-stage solver convention for orienteering problem with stochastic weights was expanded with two additional recourse actions (skip task and request replan). This allowed the vehicle to exhibit a more nuanced response when encountering energy limitations without relying on communication with an external operator or planner. These additional recourse actions could be adopted and extended into the OR field for application in other problem domains where communication is a challenge. New solvers could be developed based on these strategies that produce more optimal solutions across many fields of application (e.g. transport, logistics, military operations), changing the way industries and infrastructure operates for the better.

5.4 Recommendations for Future Work

The energy planner framework proposed in this thesis is based on formulating the AMV mission as a specific optimisation problem (as is the style within the OR field). Extending the energy

planner means adjusting the optimisation problem to deal with more complicated tasks (i.e. removing the non-hierarchical assumption described in assumption 1) and with a variety of vehicles (assumption 2). A hierarchical representation of the mission could be achieved by embedding required tasks within a dependent task's I_t (defined in 1), and then only including task into the search space if the dependencies have been included in an individual particle's position.

As suggested in section 3.4, redefinition of the E matrix to be $\mathbb{R}^{N_T \times N_T \times N_V}$ allows multiple vehicle models to be considered by the planner. Vehicle capabilities for completing specific tasks can be considered by extending P as a N_V long set of control vectors that reference $\{T_{P_i} \in \mathcal{T} \mid 1 \leq i \leq N_V, i \in \mathbb{N}\}$, essentially defining the set of tasks in $\mathcal{M}_O$ that each vehicle is capable of doing according to $I_v \in V$.

With consideration of the energy modelling, variability exists not just in the vehicle's hydrodynamic model, but also in the environment model, the drivetrain efficiency and the electrical efficiency of the battery. The hydrodynamic and environment model variability also affect the vehicle's control error, which influences the achieved trajectory and the total energy spent. Fully simulating all of these variables to acquire the complete assessment of variability in the plan is computationally expensive to do. Future work involving energy solvers will need to meet the challenge of adequately simulating the variability of the mission plan within pragmatic limits.

The use of energy modelling with real vehicle deployments also presents the opportunity for the energy planner to learn from previous mission outcomes to produce better energy estimations for future missions. For example, given a mission profile, the planner could look-up previously completed missions with similar profiles and use the collected energy consumption data as reference. A machine learning algorithm could then modify the random variable parameters used by the Monte Carlo simulation, using the error in energy cost between predicted and historical as the loss variable. Such an implementation would allow the planner to develop its own model of the vehicle's energy profile for given missions, but is challenged with appropriately representing the mission profile and energy profile so that an operator can interpret and provide human verification.

Planning with uncertainty is a challenge both in AI and OR fields. AI planning has presented a generalised approach to the domain of AMVs, allowing operators to expand the domain definition with specific vehicle models, constraints and tasks according to their applications. On the other hand, OR planners for AMVs are specialised use-case solutions that allow efficient

plans to be generated for specific scenarios. Because AI planning is flexible and is able to be adapted quickly to end user specifications, the state of AI planners is maturing faster than OR implementations. In this respect, there is an opportunity to enhance existing AI planning frameworks for AMVs through the development and integration of an energy modelling module. In the short-term this would allow the already effective temporal-based AI planners to consider energy predictions as a secondary resource constraint. In the long-term, temporal representation of missions could be swapped out entirely for energy representation, kept only as a convenient and understandable visualisation tool for operators.

References

Ferreira, A. S., Costa, M., Py, F., Pinto, J., Silva, M. A., Nimmo-Smith, A., Johansen, T. A., de Sousa, J. B., and Rajan, K. (2018). Advancing multi-vehicle deployments in oceanographic field experiments. *Autonomous Robots*.